STATUS OF THESIS

Title of thesis	DEEP LEARNING APPROACH FOR DETECTION AND CLASSIFICATION OF COLONIC POLYP

I WIN SHENG LIEW

hereby allow my thesis to be placed at the Information Resource Center (IRC) of Universiti Teknologi PETRONAS (UTP) with the following conditions:

1. The thesis becomes the property of UTP
2. The IRC of UTP may make copies of the thesis for academic purposes only.
3. This thesis is classified as

☐ Confidential

☑ Non-confidential

If this thesis is confidential, please state the reason:

_____

_____

The contents of the thesis will remain confidential for _____ years.

Remarks on disclosure:

_____

_____

Signature of Author

Permanent address:

277, Taman Wan San, 31100

Sungai Siput (U), Perak.

Date : 20-02-2022

Endorsed by

On behalf of Dr Cheng Kai Lu

Signature of Supervisor

Name of Supervisor

Cheng-Kai Lu

Date : 21-02-2022

UNIVERSITI TEKNOLOGI PETRONAS

DEEP LEARNING APPROACH FOR DETECTION AND CLASSIFICATION OF COLONIC POLYP

By

WIN SHENG LIEW

The undersigned certify that they have read, and recommend to the Postgraduate Studies Programme for acceptance of this thesis for the fulfillment of the requirements for the degree stated.

Signature: On behalf of Dr Cheng Kai Lu

Main Supervisor: Dr. Cheng-Kai Lu

Signature: ASSOC PROF DR TANG TONG BOON Director Institute of Health and Analytics for Personalised Care Research & Innovation University Teknologi PETRONAS

Co-Supervisor: AP. Dr. Tang Tong Boon

Signature:

Chair of Department: AP. Dr. Mohamad Naufal bin Mohamad Saad

Date: 22/02/2022

DEEP LEARNING APPROACH FOR DETECTION AND CLASSIFICATION OF COLONIC POLYP

by

WIN SHENG LIEW

A Thesis

Submitted to the Postgraduate Studies Programme

as a Requirement for the Degree of

MASTER OF SCIENCE

ELECTRICAL & ELECTRONICS ENGINEERING

UNIVERSITI TEKNOLOGI PETRONAS

BANDAR SERI ISKANDAR,

PERAK

FEBRUARY 2022

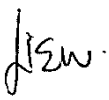
DECLARATION OF THESIS

Title of thesis	DEEP LEARNING APPROACH FOR DETECTION AND CLASSIFICATION OF COLONIC POLYP

I WIN SHENG LIEW

hereby declare that the thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at UTP or other institutions.

Witnessed by



Signature of Author

Signature of Supervisor

Permanent address:
277, Taman Wan San, 31100
Sungai Siput (U), Perak.

Name of Supervisor
Cheng-Kai Lu

Date : 20-02-2022

Date : 21-02-2022

DEDICATION

Research is creating new knowledge. In this challenging research requires self-efforts as well as advice of family, colleagues, and teachers especially to my supervisor and co-supervisor. Without their affection and guidance, I am not able to get such success and honor.

ACKNOWLEDGEMENTS

This work was supported by Ministry of Higher Education, Malaysia, under Grant FRGS/1/2020/TK0/UTP/02/23.

ABSTRACT

The increased incidence of colorectal cancer (CRC) and its mortality rate have attracted interest in the use of artificial intelligence (AI) based computer-aided diagnosis (CAD) tools to detect polyps at an early stage. Although these CAD tools have thus far achieved a good accuracy level to detect polyps, they still have room to improve further (e.g., sensitivity). Besides, the physicians need to manually detect and segment the poylps during the endoscopic screening. This process is affected by physicans' subjectivity, i.e., attention and practical experience, and it is time consuming. Moreover, there is the chances of polyps miss-detected due to the inexperience or junior physicians.Therefore, in this thesis, we propose a novel approach to distinguish colonic polyps by integrating several techniques, including a modified deep residual network, principal component analysis, and AdaBoost ensemble learning. A powerful deep residual network architecture, ResNet-50, was investigated to reduce the computational time by altering its architecture. To keep the interference to a minimum, median filter, image thresholding, contrast enhancement, and normalisation techniques were exploited on the endoscopic images to train the classification model. Three publicly available datasets, i.e., Kvasir, ETIS-LaribPolypDB, and CVC-ClinicDB, were merged to train the model, including images with and without polyps. The proposed approach achieved Matthews Correlation Coefficient of 0.9819 with accuracy, sensitivity, precision, and specificity of 99.10%, 98.82%, 99.37%, and 99.38%, respectively. However, most AI models are implemented on the software platforms. Along with the demands of embedded devices, the hardware implementation can fulfill the demands of real-time applications with high accuracy and low-power comsumption. To determine the feasibility for the convolutional neural network (CNN) to be implanted in an embedded device, we thus propose a 4-layers model to be implanted in the microprocessor. The essential functions in the CNN (i.e., padding, convolution, ReLU, max-pooling, fully-connected, and softmax) have been implemented in the microprocessor.

ABSTRAK

Peningkatan insiden kanser kolorektal dan kadar kematiannya telah menarik minat dalam penggunaan alat CAD (Computer-Aided Diagnosis) melalui AI (Artificial Intelligence) untuk mengesan polip pada peringkat awal. Walaupun alat CAD ini setakat ini telah mencapai tahap ketepatan yang baik untuk mengesan polip, ia masih mempunyai ruang untuk penambahbaikan (cth., sensitiviti). Selain itu, doktor perlu mengesan dan segmen polip secara manual semasa pemeriksaan endoskopik. Proses ini dipengaruhi oleh subjektiviti doktor, iaitu, perhatian dan pengalaman praktikal, dan ia memakan masa. Tambahan pula, terdapat kemungkinan polip tidak dapat dikesan kerana kurang pengalaman atau doktor muda. Oleh itu, dalam tesis ini, kami mencadangkan pendekatan baru untuk membezakan polip kolon dengan menyepadukan beberapa teknik, termasuk rangkaian sisa dalam yang diubah suai, analisis komponen utama, dan AdaBoost pembelajaran ensemble. Senibina rangkaian sisa dalaman yang berkuasa, ResNet-50, telah disiasat untuk mengurangkan masa pengiraan dengan mengubah senibinanya. Untuk memastikan interferansi pada tahap minimum, penapis median, ambang imej, peningkatan kontras dan teknik penormalan telah dieksploitasi pada imej endoskopik untuk melatih model klasifikasi. Tiga set data yang tersedia secara terbuka, iaitu, Kvasir, ETIS-LaribPolypDB dan CVC-ClinicDB, telah digabungkan untuk melatih model, termasuk imej dengan dan tanpa polip. Pendekatan yang dicadangkan mencapai MCC sebanyak 0.9819 dengan 99.10%, 98.82%, 99.37%, dan 99.38% bagi ketepatan, kepekaan, ketepatan dan kekhususan. Walau bagaimanapun, kebanyakan model AI dilaksanakan pada platform perisian. Bersama dengan permintaan peranti terbenam, pelaksanaan perkakasan boleh memenuhi permintaan aplikasi masa nyata dengan ketepatan tinggi dan penggunaan kuasa rendah. Untuk menentukan kebolehlaksanaan CNN untuk ditanam dalam peranti terbenam, kami mencadangkan model 4 lapisan untuk ditanam pada mikropemproses. Fungsi utama dalam CNN (cth., padding, konvolusi, ReLU, max-pooling, fully-connected, dan softmax) telah dilaksanakan pada mikropemproses.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AI	Artificial Intelligence
ACC	Accuracy
AUC	Area under the Curve
AUROC	Area Under the Receiver Operating Characteristic
CAD	Computer-Aided Diagnosis
CCS	Code Composer Studio
CNN	Convolutional Neural Network
CRC	Colorectal Cancer
CV	Cross Validation
DAP	Data Analysis Protocol
DL	Deep Learning
DMA	Direct Memory Access
DSP	Digital Signal Processor
FC	Fully-Connected
FE	Feature Extractor
FLOPs	Floating Point Operations
FN	False Negative
FP	False Postitve
FPGA	Field Programmable Gate Array
FPR	False Positive Rate
FRAM	Ferroelectric Random-Access Memory
GI	Gastrointestinal
ICA	Independent Component Analysis
IDE	Integrated Development Environment
MAC	Multiply-and-Accumulate
MAQC	MircoArray Quality Control
MCC	Matthews Correlation Coefficient
ML	Machine Learning

MPY	Multiplier
MRI	Magnetic Resonance Imaging
PCA	Principal Component Analysis
PE	Processing Engine
PRE	Precision
ReLU	Rectified Linear Unit
ROC	Receiver Operating Characteristic
SEN	Sensitivity
SEQC	Sequencing Quality Control
SGDM	Stochastic Gradient Descent with Momentum
SoC	System on Chip
SPEC	Specificity
SRAM	Static Random-Access Memory
SSD	Single-Shot Detector
SVM	Support Vector Machine
TFL	Transfer Learning
TN	True Negative
TP	True Positive
TPR	True Positive Rate
t-SNE	t-Stochastic Neighbourhood Embedding
UMAP	Uniform Manifold Approximation and Projection

LIST OF SYMBOLS

σ	Standard deviation / activation
x	Input
$\bar{x}$	Mean of feature vector
b	Bias
ω	Weights (network parameter)
ε_k	Error rate
D_k	Weight distribution
Z_k	Normalisation factor
α_k	Weight coefficient
$f_k(x)$	Number of weak classifier
$Y(x)$	Final classifier

CHAPTER 1

INTRODUCTION

This chapter introduces colorectal cancer (CRC) with an overview of the state of case and mortality rate of this disease worldwide. Besides, it discusses the existing computer-aided diagnosis (CAD) and medical capsule robots in assisting the physician to carry out the diagnostic. This chapter also reviews the current issues that cause the increasing cases and its mortality rate, which leads to several hypothesis and research questions in solving this matter before coming up with the research objectives to fill those research gaps. At the end of the chapter, scope of study is presented.

1.1 Research Background

CRC is the third most common malignancy and the fourth leading cause of cancer death in the world [1]. An analysis from the American Cancer Society showed that both the number of new CRC cases and the mortality rate have been increasing [1]. CRC begins in the form of glandular tissue, known as a polyp, on the inner lining of the colon or rectum [2-4]. Untreated neoplastic polyps may turn into CRC. Therefore, detecting polyps and removing them early can greatly reduce the incidence of CRC and its mortality rate [6-8].

Computer-aided diagnosis (CAD) has been one of the most reliable and widely used methods in screening, medical diagnosis, and therapeutic systems for various cancer diseases, including CRC, over the past decade. These CAD systems help physicians to focus on smaller sub-volumes instead of on the entire volume, which significantly helps physicians make accurate decisions regarding the removal of polyps at an early stage, which, in turn, is beneficial for curative interventions [9-11]. Recently, the use of artificial intelligence (AI) techniques, such as deep learning (DL) and machine learning (ML), have opened the door for the use of CAD in the

interpretation of medical images, helping physicians provide a diagnosis by acting as a second reader in detecting cancer diseases. In this case, the AI-based models can assist physicians in performing detection and characterizing lesions seamlessly while manipulating an endoscope and efficiently interpreting the endoscopic images [12].

Usually, an image classification model is divided into three parts: image pre-processing, feature extraction, and classification. Due to the quality of the images, all images need to undergo pre-processing to reduce noise and degradation. Most of the previous computer-assisted polyp classification systems were based on handcrafted feature extraction methods to train a primary classifier [3]. The ML algorithms can learn the classification part; however, the feature extraction part still requires expert engineering support from a human being [13]. DL has shown remarkable results for image classification in computer vision, with extraordinary accuracy [14-17]. In DL, the convolutional neural network (CNN) is a very powerful ML technique and part of deep neural networks. DL can also easily extract higher level and more abstract features [18,19], unlike the early, conventional handcrafted methods; CNN features outperformed handcrafted features in the 2015 Endoscopic Vision Challenge [20]. Hereby, CNNs can automatically learn rich feature representations from many diverse images to carry out the classification task [4]. Over the years, the performance of the CNN has been significantly improved by employing depth and other structural modifications, such as the block architectures [19]. These blocks are the auxiliary learners that allow boosting CNN performance by making problem-aware learning [19].

Classification is a type of supervised ML technique that is used to predict a discrete class where the classes are predefined for each event [21]. In ML, the ensemble method is mainly utilised to enhance a classifier's efficiency [21], and it combines various learning algorithms/classifiers to categorise new samples to obtain better predictive accuracy [22]. The most common types of ensemble methods include bagging, boosting, stacking, and voting. In previous studies [23-25], some basic ML algorithms such as support vector machine (SVM), decision tree, and Naïve Bayes were utilised for disease classification, but the ensemble techniques, such as bagging and boosting,

can improve the accuracy of classification significantly by combining multiple weak classifiers.

Nevertheless, most CNN models are trained and implemented on the software platforms [26]. It is a considerable challenge to implant the CNN architectures, which consists of many neurons, into embedded devices with a battery supplied. In the wake of demands of embedded devices, the hardware implementation of CNNs can fulfill the demands of real-time applications with high accuracy and low power consumption. In particular, the hardware implementations can be used as an embedded device to perform some specific biomedical disease tasks such as stress detection, seizure detection, stroke detection, etc. For colon polyp detection, medical capsule robot could be the embedded device to capture the images of the GI tract. The development of medical capsule robots has emerged from the science fiction notion of robots travelling inside the gastrointestinal (GI) tract to perform diagnosis and treatment [27]. Today, varieties of capsule robots are available in the market with diagnostic features, such as in vivo body temperature and pH monitoring [27], and yet these are not the main function for a capsule robot during polyp detection. In this case, a capsule robot with the feature of automatically detect and localise polyp would be more important. Therefore, a CNN-based solution in the smart capsule robots that will detect polyps automatically can assist physicians in performing detection more precisely and effectively.

1.2 Problem Statement

Endoscopy/colonoscopy is a practical way to discover and detect colonic polyps. Nonetheless, the common issues faced in this research are listed as follows:

a) On the conventional colonoscopy, the physicans need to manually detect and segment the polyps. The procedure is affected by physicians' subjectivity, such as attention and practical experience, which caused time consuming [28].

b) Many existing CADs have achieved good results in classifying colon polyps, but they still have room to improve further, especially when high sensitivity is

an important concern. That is because the cost of false negatives (i.e., missed polyps) is much higher than false positives [29].

c) Due to the subjectivity of physicians, it was reported the miss-rate for polyp detection is about 22% during the endoscopy/colonoscopy screening [5].

1.3 Research Questions

To solve the problems and achieve the hypothesis, the research questions come out:

a) Could the application of AI be developed into the CAD tool to perform an automated detection and classification of colonic polyp ?

b) Could the performance of CAD tool be further improved by using a new combination of modified deep residual convolutional neural network, feature dimension reduction and AdaBoost techniques?

c) Could the CNN-based algorithm be implemented on the microprocessor to achieve the idea of integrating system on chip (SoC) on the camera of medical capsule robot?

1.4 Research Hypothesis

The mainstream in this study is to develop a competitive and robust method in detecting the colonic polyps to reduce the cases of CRC and its mortality rate. It is hypothesized that the research gaps can be filled accordingly:

a) CAD tool with AI-based helps the physician in performing detection and characterizing lesions seamlessly, which acts like a second reader in detecting polyps.

b) The ability to further enhance the performance of existing CAD tools allows the physicians to provide a more accurate and is with high sensitivity diagnosis to the patients.

c) Smart medical capsule robot aids in automatically detect and classify the polyps when travelling inside the GI tract.

1.5 Research Objectives

The overall aim of the research is to help the physicians by proposing an automatic method in analyzing endoscopic images. It can be accomplished with the following objectives:

a) To develop an automatic method to detect and classify colonic polyps using the combination of image processing and AI, such as DL and ML.

b) To improve the performance of existing CAD tool for colonic polyps detection in terms of MCC, accuracy, sensitivity, specificity, and etc.

c) To determine the feasibility for the CNN to be implanted in an embedded device, which is the medical capsule robot with the feature of automatically detect and classifiy colonic polyps.

1.6 Scope of Research

The focus of this study is to i) delovop a novel approach that able to detect the presence of abnormalities from endoscopic images, ii) improve the current CAD tools, and iii) develop a smart medical capsule robot through DL technique. The study divides into two phases to comply with the research objectives, where phase I subject for the abnormality detection and performance improvement, while phase II for the implementation of embedded device. A MATLAB computing tool was used to develop the algorithm for phase I. For phase II, an integrated development environment (IDE), Code Composer Studio (CCS) was utilized to develop the algorithm into the microprocessor.

Three online publicy datasets (i.e., Kvasir, ETIS-LaribPolypDB, and CVC-ClinicDB) experiment on both phases. The final results for both phases were evaluated

based on its Matthews Correlation Coefficient (MCC), accuracy, sensitivity, precision, specificity, and area under the curve (AUC). The aforementioned datasets consist of two classes of images (polyps and normal colon), they were merged to obtain a larger database to train and develop a competitive CAD system and a smart medical capsule robot.

1.7 Thesis Organization

The thesis is organized into five chapters starting from introduction, followed by related works, methodology, results and discussion, and lastly, wrapped with conclusion. The description of the remaining chapters is presented as the following:

Chapter 2 discussed the previous works done on DL for colonic polyp detection and classification. Besides, the previous works on the hardware implementation of CNN for various detection tasks were reviewed. Several methods used for each operation have been discussed with their performance, advantage, and limitation. Lastly, a critical review was discussed based on the literature review.

Chapter 3 presented the algorithm used in each process,including the explanation and discussion on the chosen technique adapted for both phases. This chapter begins with a brief description of the chosen datasets, followed by the explanations on the proposed method for the CAD tool, mathematical assessments for the performance evaluation, and lastly, the discussion on the proposed method for the hardware implementation.

Chapter 4 explained obtained results from the proposed method introduced in the preceding chapter. Furthermore, the performance comparison between the obtained result.s and previous related works was provided in this chapter. In addition, the results for hardware implementation of CNN were presented at the end of this chapter.

Chapter 5 concluded the performance of the proposed method with descriptions of the contribution of the research work. Besides, the limitation and recommendation work for future were discussed at the end of the chapter.

CHAPTER 2

RELATED WORK

A brief implementation of DL and ML in previous CAD systems for detection and classification of colonic polyp was overviewed to clasp the idea of the application. Besides, the liteture reviews have been done on the previous works on the hardware implementation of CNN for different detection and classification tasks. In addition, the advantages and limitations of differenct techniques applicable to the CAD systems and the hardware implementation were discussed.

2.1 Existing CAD Systems for Colonic Polyp Detection

The promising performance of DL and its influence have driven the field of histological image analysis for the early diagnosis of CRC. Zhang et al. [30] used the transfer learning (TFL) approach to transfer the low-level features learned from a source domain (non-medical) to a target domain (endoscopic images). The extracted features were then fed into a SVM for the classification of colorectal polyps, resulting in an accuracy of 85.90% with a recall and precision of 87.60% and 87.30%, respectively. Additionally, Liu et al. [31] trained a deep CNN, faster_rcnn_inception_resnet_v2 model, for polyp and adenoma classification, which contained four main blocks to process the images for the prediction. First, the inception block was employed to extract the feature map and sent for the first round of rough prediction. Simultaneously, it was sent to another block to extract the features for the second stage of prediction, where the second prediction was a quadratic regression, as was the case in the first rough prediction. They obtained a mean average precision of 90.65% when the intersection over union was set to 0.5.

In a previous study [32], Bour et al. utilised a TFL approach based on ResNet-50 by changing the classifiers while keeping the same convolutional base. The ResNet-

50 model performed the best with their data, with an accuracy, recall, precision, F1-score, and specificity of 87.10%, 87.10%, 87.10%, 87.10%, and 93.00%, respectively, as compared to other models, such as ResNet-101, Xception, VGG-19, and Inception_V3. Furthermore, Patino-Barrientos et al. [3] proposed a DL model based on Kudo's classification scheme using a fine-tuned VGG network. The fine-tuned VGG model resulted in better performance, with an accuracy, recall, precision, and F1-score of 83%, 86%, 81%, and 83%, respectively, as compared to the models that trained with and without using the original VGG base model as a feature extractor. In a different study [33], Park et al. proposed a CNN model that consisted of 43 convolutions with one fully-connected layer for the classification of colonoscopy images. Their proposed network had fewer parameters, which made the network less complex; meanwhile, it achieved an accuracy of 94.39% in the test result, which is higher than VGG-19, ResNet, and DenseNet when trained on their dataset.

Ozawa et al. [34] used a deep CNN architecture, namely the Single Shot MultiBox Detector, to detect and classify colorectal polyps through endoscopic images; their trained CNN achieved a sensitivity of 92% and a positive predictive value of 86%. Patel et al. [4] compared the performance of six CNN models (VGG-19 with and without batch normalisation, ResNet-50, DenseNet, SENet, and MnasNet) for polyp detection. However, in that study, the more advanced models, like ResNet-50, DenseNet, SENet, and MnasNet, did not perform well; VGG-19 achieved the best result. Moreover, Wittenberg et al. [35] exploited the Mask R-CNN architecture with ResNet-101 to extract the image features. Due to the small size of datasets, they also used the TFL approach to train a complex network architecture and achieved a sensitivity of 87% and F1-score of 83.33%. Wang et al. [36] combined the classical CNN models, VGG and ResNet, with global average pooling. All these models obtained an accuracy of above 98%, and the true negative rate and true positive rate were above 98% and 96%, respectively. The experimental results showed that their proposed approach not only achieved high classification accuracies, but also reduced the network's parameters, making the model lightweight.

Liu et al. [37] have proposed a single-shot detector (SSD) framework with InceptionV3 as a feature extractor, in which the SSD uses a feed-forward CNN to

create a fixed-size boundary box for each object on different feature maps. The SSD model has obtained sensitivity and F1-score of 80.30% and 76.80%, respectively. Moreover, Zhou et al. [38] developed a dense convolutional network, CRCNet, for the optical diagnosis for CRC. In CRCNet, the network connects in a feed-forward manner, which improves the flow of information and feature exploration. Vani and Mahendra [39] conducted an analysis to compare the performance of different DL techniques. In their research, VGG-19 achieved the best with an accuracy, F1-score, and sensitivity of 94.45%, 93%, and 94%, respectively. Lee et al. [29] utilised YOLOv2 for polyp detection, and they applied median filtering to reduce the number of false positives during the video analysis. The algorithm was validated with four independent datasets and achieved a sensitivity above 87%.

In a previous study [40], Nadimi et al. used an optimized ZF-Net algorithm with stochastic gradient descent with momentum (SGDM), which combines data augmentation, pre-processing, and TFL techniques for their colorectal polyp detection. Their algorithm achieved an accuracy, sensitivity, and specificity of 98%, 98.10%, and 96.30%, respectively. Wei et al. [41] exploited ResNet with weight initialisation to classify colorectal polyps on histopathologic slides. For the internal evaluation, their model had a mean accuracy of 93.50%; this was 87% for the external evaluation. In another study [42], Meng et al. used Mask R-CNN architecture with a modified version of ResNet (Res2Net) as backbone for the detection and segmentation of colorectal polyps. In this way, the bottleneck structure of ResNet was improved. Thus, their proposed framework achieved a mean average precision of 89.50%.

2.2 Hardware Implementation of CNN

CNNs have been well-implemented on many software platforms [26]. However, due to the large computation and complex structure of CNN and frequent memory access, it is difficult to implement on the embedded platform [43]. Heller et al. [44] used a low-power microcontroller MSP430FR series for seizure detection. On a dataset of 22 patients, using 56% of available runtime and 10% of available memory, they obtained a median sensitivity of 100%, false-positive rate (FPR) of 20.7fp/h, and a shorter detection delay of 2.7s as compared to Hügle et al. [45], which it is suitable for the

application in an implantable closed-loop device. The CPU runtime was minimized and the power consumption was reduced to 802 µW, by efficiently utilisating the available hardware modules. However, the runtime of the microcontroller is a poor criterion for the estimation of power conscumption, a highly dynamic power consumption is expected.

Hügle et al. [45] presented a CNN for the early seizure detection on a low-power microcontroller MSP430FR. Their proposed method achieved a median sensitivity of 0.96, FPR of 10.1fp/h, and a median detection delay of 3.7s. Their research compared with Kiral-Kornek et al. [46], which used IBM TrueNorth chip, they computed a power consumption of 850µW, which is 5 to 8.8 times lower for the preprocessing and forward pass of SeizureNet. Nevertheless, the detection of electroencephalographic seizure patterns occurs at the cost of higher FPR, which causes increases the chances of wrongly classify artefacts as ictal patterns. Odagawa et al. [47] proposed a CNN and machine learning with Tensilica® Xtensa Vision P6 digital signal processor (DSP) on Protium S1Field Programmable Gate Array (FPGA) for the colorectal tumor detection. To optimize the preprocessing of input images, the DSP is utilised as it has specific vector instructions and libraries for efficient load and sore to memories. Their prototyped system is able to classify the existence of cancer in the lesion and obtain a real-time image processing on 30fps at 200MHz.

Khatwani et al. [48] deployed their energy-efficient CNN on the Artix-7 FPGA for electroencephalography artifact detection. In their research, the reason for the chosen FPGA is that its on-chip memories are sufficient to store the CNN model and the intermediate data. Besides, 4 number of processing engines (PEs) at 11.1MHz is the optimal configuration to meet the lowest energy and power consumption. Their FPGA with 4-PE implementation outperformed the low power configuration of TX2 by 65× and 2× in terms of power efficiency and energy efficiency, respectively. Compared with Jafari et al. [49], their FPGA implementation outperformed the Independent Component Analysis (ICA) method by 11 in terms of energy efficiency on the same dataset. However, on their CNN-based model implementation, they show that the increasing number of PEs leads to an increase of power consumption and a decrease in processing latency. Naresh Gowda and Rasheed [50] proposed a hybrid classifier

on the Zync SoC FPGA platform to detect cancer cells. Their proposed system obtained a high performance and achieved an efficient classification technique to detect the cancer cell with low power consumption of 1.815W and low hardware utilization. Furthermore, Chen et al. [51] implemented a customized CNN accelerator based on the Xilinx FPGA for pulse waveform classification. In the experiment, their self-designed CNN achieved a high accuracy with fewer parameters, and the accelerator obtained a latency of 0.827ms and took only 0.714W at the working frequency of 100MHz. The optimized CNN model and the custom register-transfer-level design help improve the clock speed and reduce the latency and power consumption.

2.3 Critical Review

a) The complex CNNs such as Inception ResNet-V2, ResNet-101, and VGG-19 have a promising classification performances. However, they consist of deeper and more complex neural networks, resulting high computational cost. A good network should has a high degree of accuracy, as well as with a fast computational time.

b) The CNNs with deeper layers has a drawbacks of overfitting and huge processing time. Therefore, those deep CNNs may have a poor performance. However, it is still depending on the applications as different structures of the network are suitable for different tasks, resulting in different outcomes.

c) FPGA performs well as it is an efficient platform for prototyping and performance exploration due to its multiple multiply-accumulate units. Nevertheless, FPGA has a higher power consumption as compared to microcontroller. Besides, FPGA suffers from a small internal memory.

2.4 Summary

In essence, DL and ML play an important role in medical image analysis as they help in prediction of a positive or negative test for a specify disease. As for this research

work, several techniques were combined for the phase I algorithm to develop a competitive assisting CAD tool to detect the colonic polyp. A TFL was adopted through using CNN as feature extractor to improve the performance and the learned features were shared to the classifier for cell discrimination (normal and abnormal). Additionally, for the phase II, a novel approach was proposed to implant the CNN into a closed-loop device for the detection of colonic polyp. This approach could achieves a smart medical capsule robot that travels in the GI tract to detect the abnormabilities.

CHAPTER 3

METHODOLOGY

In this chapter, the present study highlights the classification of abnormalities, that is, the occurrence of polyps in endoscopy images, using a new combination of modified deep residual CNN with the ensemble classification technique, AdaBoost. To improve the performance of CNN, we also investigated the effects of distinct structure outcomes for DL by changing the architecture of ResNet-50. The modified ResNet-50 was exploited as a feature extractor for training an AdaBoost ensemble classifier. However, AdaBoost has some limitations as it is based on empirical evidence and is particularly vulnerable to uniform noise; the weak classifiers being too weak can lead to overfitting [52]. Thus, principal component analysis (PCA) was applied to reduce the dimension of the features before feeding into the classifier. Finally, the results were assessed based on several evaluation criteria: Matthews Correlation Coefficient (MCC), accuracy, sensitivity, precision, specificity, and area under the receiver operating characteristic (AUROC) curve. Hardware implementation for medical capsule robot that can classify lesions in real-time is the second phase of the study. To prove this idea, 4-layers CNN is introduced, all the essential functions in CNN, such as padding, convolution, ReLU, max-pooling, softmax, and fully-connected, are investigated and implemented on a microprocessor.

3.1 Proposed Methodology

The suggested methodology consists of i) an autonomous CAD system that highly competitive in detecting the presence of anomalies in the endoscopic images, and ii) a smart medical capsule robot that capable in detecting the abnormalities in the GI tract. A general overview of the research was illustrated in Figure 3.1. The process flow of the proposed algorithm for both phases were outlined as in Figure 3.2 and Figure 3.3.

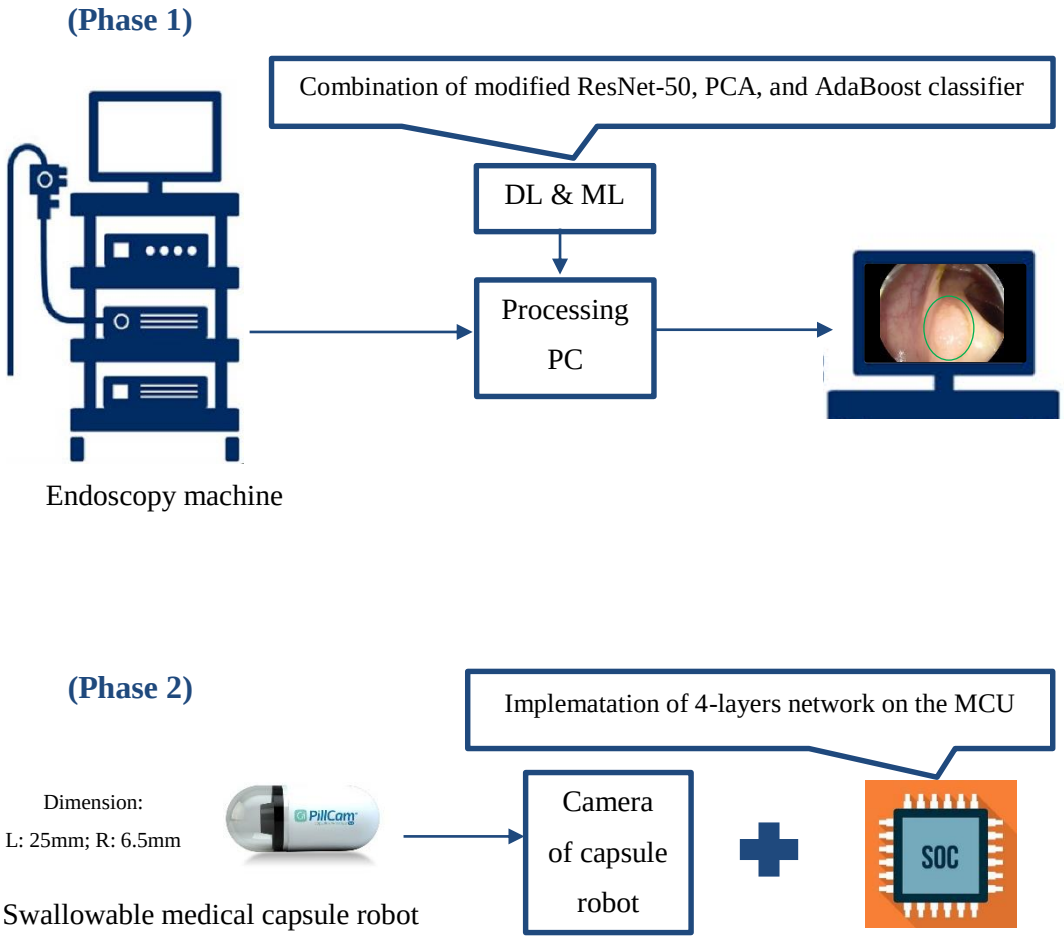


Figure 3.1: Overview of the research

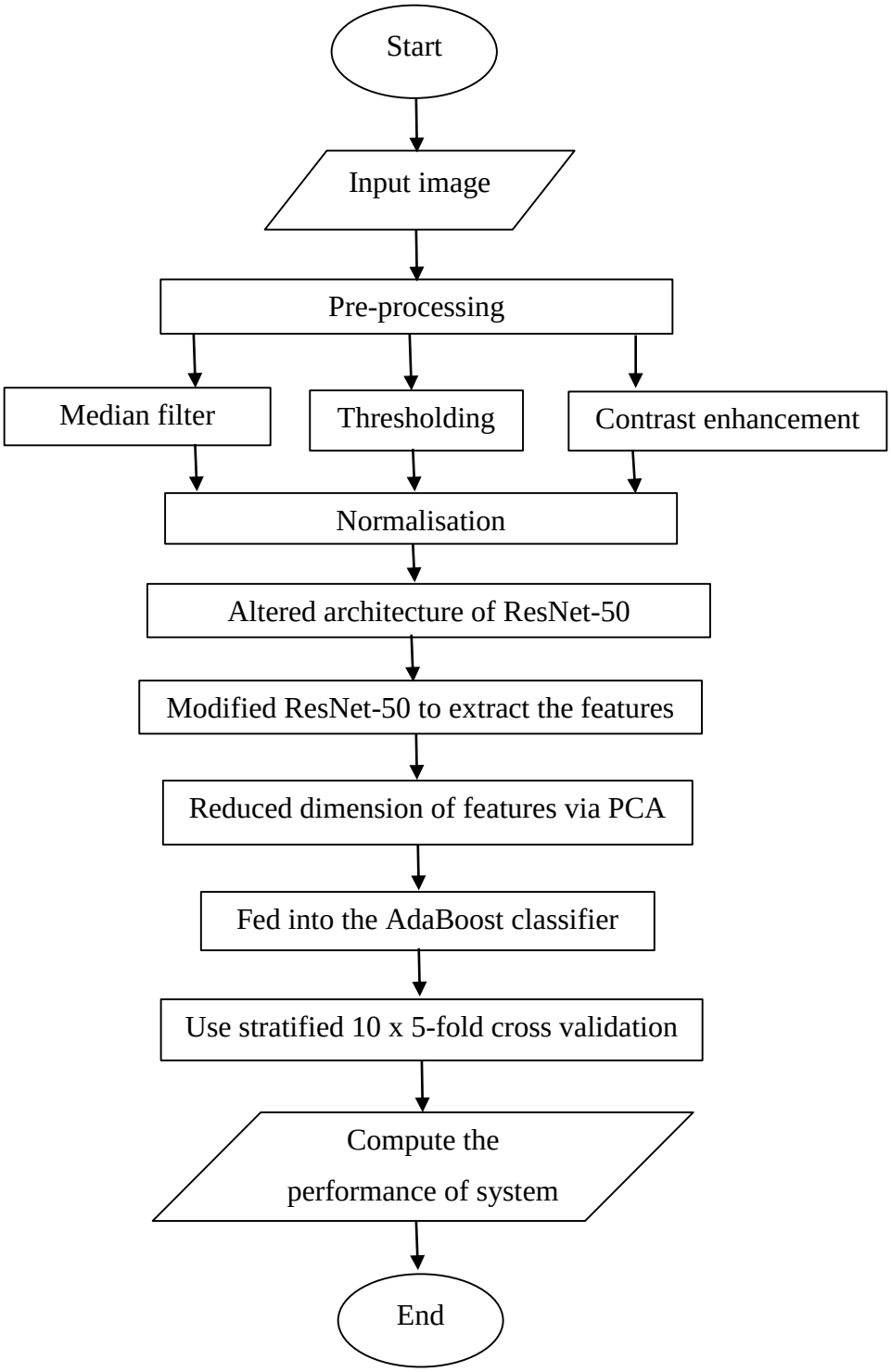


Figure 3.2: Process Flow of Phase I Algorithm

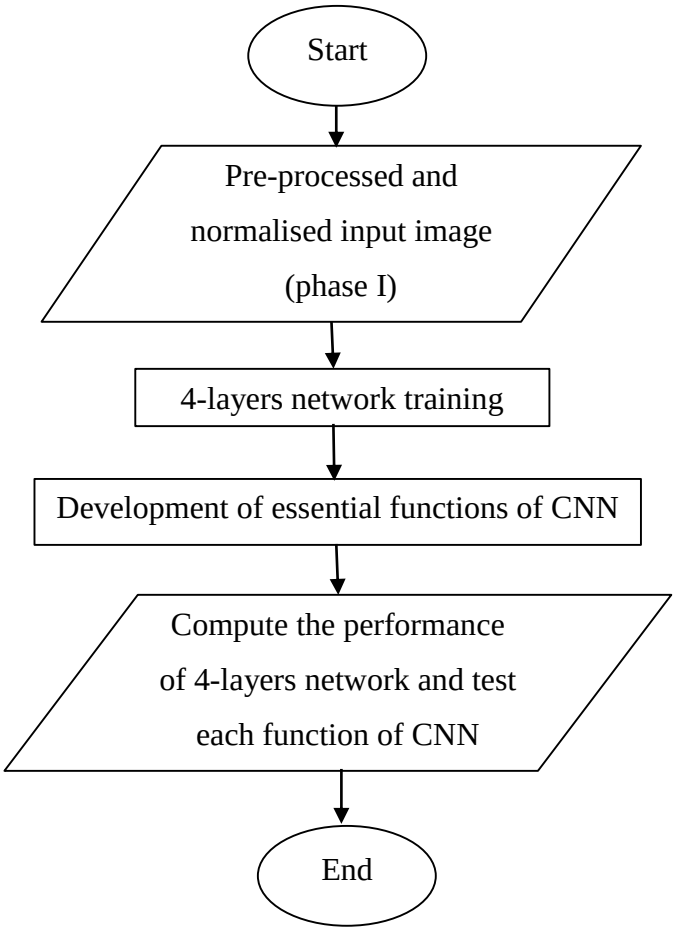
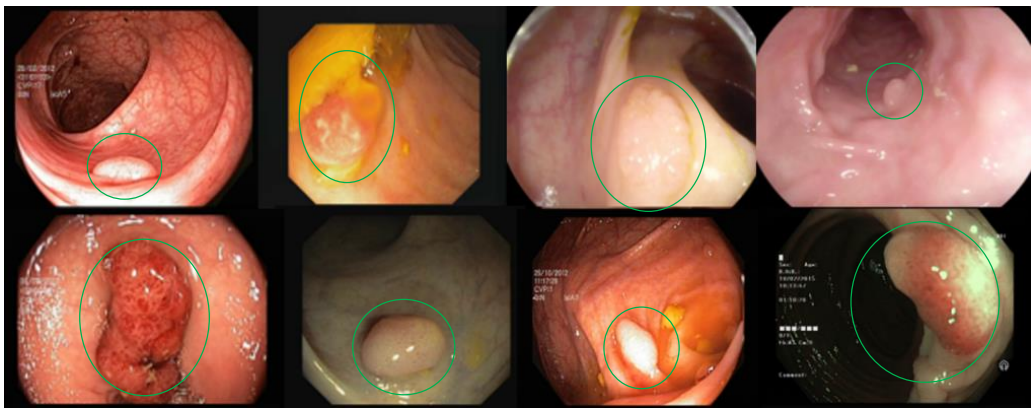


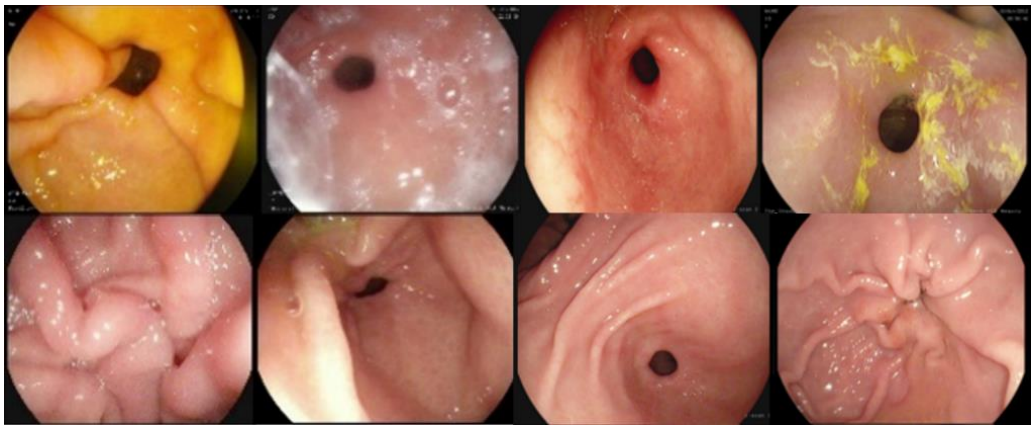
Figure 3.3: Process Flow of Phase II Algorithm

3.2 Database and Tool

Three endoscopic datasets with various features were merged to train the model. As shown in Figure 3.4, they have different visual quality and histological patterns. The details and information of each dataset were described in Table 3.1. Dataset 1 (Kvasir) was obtained from a separate study [53,54]. The Kvasir dataset was obtained through the endoscopy departments of four hospitals at Vestre Viken Health Trust (VV) in Norway. All the images were annotated by medical experts from VV and the Cancer Registry of Norway [54]. This dataset consists of eight classes of GI colour images with dimensions of 720 × 576, but only two classes (polyps and normal colon) were selected to conduct the study. This is because the rest of the classes included esophagitis, Z- line, pylorus, caecum, rectum, and dyed landmarks, which are not



(a)



(b)

Figure 3.4: Sample endoscopy images **(a)** with polyps (polyp is labelled with a green circle) and **(b)** without polyps

Table 3.1: Selected datasets used in the study. P - polyp, NP - non-polyp

Description	**Database**		
	Kvasir (Dataset 1)	**ETIS-LaribPolypDB (Dataset 2)**	**CVC-ClinicDB (Dataset 3)**
Type of data	Colour image	Colour image	Colour image
Type of format	.jpg	.tif	.tif
Data dimensions	720 x 576	1225 x 966	384 x 288
Number of data	P: 375; NP: 742	P:100	P: 300

related to this research. Hence, in total, 1117 images (375 and 742 images with and without polyps, respectively) were selected from Dataset 1 and used in this research.

Datasets 2 and 3 (ETIS-LaribPolypDB and CVC-ClinicDB) are images of polyps that were extracted from colonoscopy videos [55,56]. There are two different types of images in both datasets, including original images (colour images) and polyp masks (ground truth). However, the ground truth images were not used in this research because polyp segmentation was not the goal of our work. The images from Dataset 2 are the property of Lariboisière Hospital-APHP, France, and ETIS laboratory, ENSEA, University of Cergy-Pontoise, France; Dataset 3 is the property of Hospital Clinic, Barcelona, Spain. The datasets were composed of 100 and 300 colour images, where the data dimensions were 1225 × 966 and 384 × 288, respectively. Therefore, a total of 1517 colour images, including images with and without polyps from Datasets 1, 2, and 3, were merged to train, validate, and test the classification model on a Windows 10 64-bit operating system with an Intel ® Core i7-2600 CPU at 3.40 GHz with 16GB RAM and an AMD Radeon HD 6450 GPU. Of these, 768 were used to train the model, 329 for validation, and the remaining 420 to test the model. These tremendous diversities of images thus allowed the model to learn various types of features.

3.3 Images Pre-Processing

The lack of ambient light and the camera's optical properties mean that images captured during endoscopic screening have some noise, such as artefacts [57], vignettes [57], and illuminations [58]. Pre-processing is an important preliminary phase in the building an AI computer vision model that aims to improve the image features by suppressing these unwanted distortions and noises. In this study, we used a median filter to filter out the unnecessary information/noise from the images. The median filter [59] replaces all the image pixels simultaneously with the pre-defined (3 × 3) neighbourhood median of image pixels. Equation (3.1) below represents a generalised function for any neighbourhood:

$$f'(m,n) = med \mid (-k \leq u,v \leq k)\{F(m+u,n+v)\}, \qquad (3.1)$$

where k=1, u and v are the filter coefficient, and the median is computed over a 3 × 3 filter, leading to less noise in the image.

Furthermore, we applied image thresholding to partition an object into a foreground and background [60]. The thresholding method will automatically specify a threshold value (T), where the pixel values below T are considered the foreground and those above T are considered the background. In this work, we employed true colour images, which consist of three channels: red (R), green (G), and blue (B). To determine and find out the T for the background for each channel, an analysis was performed on the datasets, the T of the R and G channels that are below 50 and 15, respectively could be considered as background. Therefore, the T of the R and G channels were scaled to 0 if T were below 50 and 15, respectively. On the other hand, the B channel was not utilised for thresholding because the polyp and background pixel values were almost the same. In addition, the contrast of the image was enhanced by manipulating the values of colour space on lightness (L*), red/green (a*), and blue/yellow (b*) channels. To fine-tune the lightness and darkness of the images without changing the colour coordination, the contrast adjustment was done on the L* only by scaling the values to the range of 0 to 2 to preserve the original colors while keeping the a* and b* channels unchanged.

In short, the purpose of pre-processing the images is to improve the image features by suppressing these unwanted distortion and noises. After the pre-processing steps (median filter, thresholding, and contrast enhancement), image normalisation is an important step, where the dimensions of the images were scaled to 224 × 224 as input to the CNN model during training and evaluation of the model. The reason of normalising images to 224 × 224 is because it is the input dimension of ResNet-50. The input images are pre-processed to a standard normalisation, as follows:

$$x' = \frac{x - \bar{x}}{\sigma}, \tag{3.2}$$

where x is the original feature vector, $\bar{x}$ is the mean of the feature vector, and σ is its standard deviation. The images in Figure 3.5 underwent the pre-processing and normalisation processes.

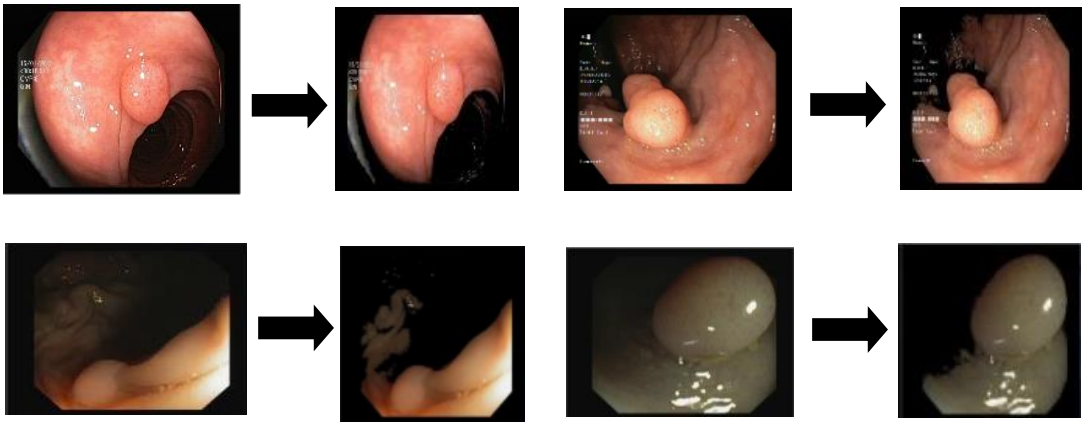


Figure 3.5: Example of images that underwent the pre-processing and normalisation operations

3.4 Ablations in the Proposed Network Architecture

The original ResNet-50 architecture had 50 deep layers. A different structure of a network results in different outcomes. In this work, we applied variants on the ResNet-50. First, to reduce the size and computational cost, the number of bottleneck residual blocks for res_block1_x, res_block2_x, res_block3_x, and res_block4_x was reduced to two, three, three, and two, respectively, and the ReLU activation between each residual_block was removed. Unlike batch normalisation, the ReLU activations disturb the data that passes through the identity connections [61]. Other than that, a new max-pooling layer was added before the bottleneck of every conv_block to keep the output size of each layer similar. These max-pooling layers have the same parameters (i.e., 3 x 3 pool size, stride 2, and padding). Besides, to keep all information without alterations, concatenations of tensors were exploited in each of the conv_block before the addition layer. Similar to InceptionNets and DenseNet, the next layer can choose to work on either the feature maps from the immediate earlier layer or the feature maps of the convolution operation before the immediate earlier layer. In Figure 3.6, the changes in ResNet-50 are shown at the image branch. The following Table 3.2 compares our modified ResNet-50 with the pre-trained networks and some of their properties. Compared to the original ResNet-50, the modified ResNet-50 has fewer filters, which means it has fewer parameters with lower complexity. Pruning of

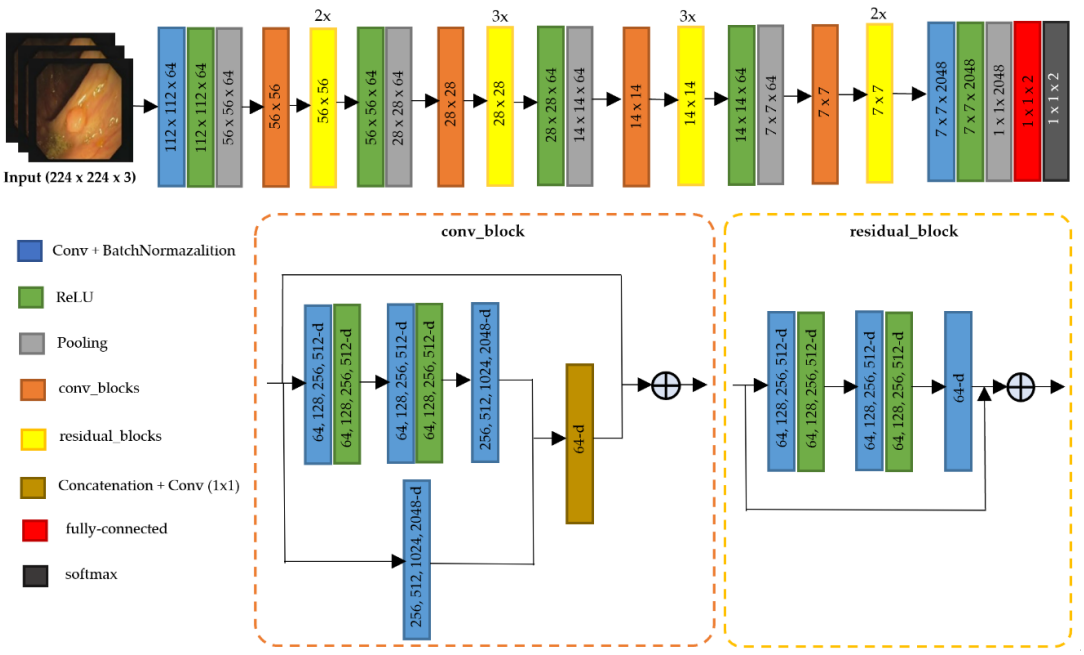


Figure 3.6: Modified ResNet-50 architecture

ResNet-50 speeds up the training and inference. Simultaneously, it retains the original performance as much as possible.

The availability of large datasets is always a major problem when building a DL model. In this case, TFL is an efficient and practical solution to overcome such problems, especially in the medical research field [62]. The employment of TFL can improve the accuracy while reducing the training time [63]. The CNNs are rebuilt by replacing the final three layers with new features specific to the image dataset of interest, with minor modifications of the original network architecture. In this research,we used some existing pre-trained networks for the TFL approach. Table 3.3 lists the properties and the performance of our proposed ResNet-50, with minor modification, and compares it with the pre-trained networks. Different structures of the network are suitable for different tasks, resulting in variable performance. Network accuracy, speed/computational time, and size are the most important characteristics when it comes to the design of a network. A good network has a high degree of accuracy with a fast computational time. After training, Inception ResNet-V2 had the highest accuracy, followed by the modified ResNet-50 and the original ResNet-50. This study showed that the modified ResNet-50 with a smaller size achieved slightly

Table 3.2: Architecture of the original and modified ResNet-50

Layer name	**Output size**	**Original ResNet-50**	**Modified ResNet-50**
conv_1	112 x 112	7 x 7, 64-d, stride 2	Same
conv_block_1	56 x 56	[1 x 1, 64-d,stride 1 3 x 3, 64-d, stride 1 1 x 1, 256-d, stride 1 1 x 1, 256-d, stride 1]	[1 x 1, 64-d,stride 1 3 x 3, 64-d, stride 1 1 x 1, 256-d, stride 1 1 x 1, 256-d, stride 1 1 x 1, 64-d, stride 1]
res_block_x	56 x 56	[1 x 1, 64-d, stride 1 3 x 3, 64-d, stride 1 1 x 1, 256-d, stride 1] x 2	[1 x 1, 64-d, stride 1 3 x 3, 64-d, stride 1 1 x 1, 64-d, stride 1] x 2
conv_block_2	28 x28	[1 x 1, 128-d,stride 2 3 x 3, 128-d, stride 1 1 x 1, 512-d, stride 1 1 x 1, 512-d, stride 2]	[1 x 1, 128-d, stride 1 3 x 3, 128-d, stride 1 1 x 1, 512-d, stride 1 1 x 1, 512-d, stride 1 1 x 1, 64-d, stride 1]
res_block_2	28 x28	[1 x 1, 128-d, stride 1 3 x 3, 128-d, stride 1 1 x 1, 512-d, stride 1] x 3	[1 x 1, 128-d, stride 1 3 x 3, 128-d, stride 1 1 x 1, 64-d, stride 1] x 3
conv_block_3	14 x14	[1 x 1, 256-d, stride 2 3 x 3, 256-d, stride 1 1 x 1, 1024-d, stride 1 1 x 1, 1024-d, stride 2]	[1 x 1, 256-d, stride 1 3 x 3, 256-d, stride 1 1 x 1, 1024-d, stride 1 1 x 1, 1024-d, stride 1 1 x 1, 64-d, stride 1]
res_block3_x	14 x14	[1 x 1, 256-d, stride 1 3 x 3, 256-d, stride 1 1 x 1, 1024-d, stride 1] x 5	[1 x 1, 256-d, stride 1 3 x 3, 256-d, stride 1 1 x 1, 64-d, stride 1] x 3
conv_block_4	7 x 7	[1 x 1, 512-d, stride 2 3 x 3, 512-d, stride 1 1 x 1, 2048-d, stride 1 1 x 1, 2048-d, stride 2]	[1 x 1, 512-d, stride 1 3 x 3, 512-d, stride 1 1 x 1, 2048-d, stride 1 1 x 1, 2048-d, stride 1 1 x 1, 64-d, stride 1]
res_block4_x	7 x 7	[1 x 1, 512-d, stride 1 3 x 3, 512-d, stride 1 1 x 1, 2048-d, stride 1] x 2	[1 x 1, 512-d, stride 1 3 x 3, 512-d, stride 1 1 x 1, 64-d, stride 1] x 2
conv_2	7 x 7	-	1 x 1, 2048-d, stride 1
avg_pool	1 x 1	2048-d	2048-d
fc	1 x 1	1000-d	2-d
softmax	1 x 1	1000-d	2-d

Table 3.3: Comparison of the proposed ResNet-50 with pre-trained networks [66,67]

Network	Number of layers	Size (MB)	Parameters (millions)	Training accuracy (%)	Computational time (min)
Modified ResNet-50 (proposed)	164	21	5.7	96.96	161
GoogLeNet	144	27	7.0	94.68	67
ResNet-50	177	96	25.6	96.94	209
ResNet-101	347	167	44.6	96.73	288
Inception ResNet-V2	824	209	55.9	97.50	953
AlexNet	25	227	61.0	51.37	22
VGG-19	47	535	144.0	96.28	492

better performance than the original ResNet-50. However, in terms of computation time, a deep and complex network, such as Inception ResNet-V2, requires both a longer processing time and a longer training time [64], yet the computation time depends on the size of the neural network [65] and the amount of data. In this work, the modified ResNet-50 was utilised since we took into consideration the high computational complexity of deeper and more complex neural networks (i.e., ResNet-101, VGG-19, and Inception ResNet-V2) that need to be developed in the hardware in future work.

3.5 Feature Extraction and Dimension Reduction

In ML and Image processing, feature extraction interprets the relevant information in an image so that the latter classification task is made easy by a formal procedure [68]. Based on the content of the object in the image, the derived values (features) are built for every image. In this work, an effective DL-based feature extraction was used.

As shown in Figure 3.7, the input data flow into the CNN contains couples of convolutional layers with pooling layers. Furthermore, using a deeper layer for feature extraction improves the classification performance compared to the shallower layers because deeper layers contain higher-level features built using the lower-level features of earlier layers. These higher-level features contain of histologic features and morphologic features. The histologic features include tubular, villous, and tubulovillous; while morphology features consist of colour, shape, texture, sessile, pedunculated, and flat.

A huge collection of different images can be trained using CNNs since CNNs have the ability to learn rich feature representations for a vast number of images [4]. These feature representations often transcend the conventional handcrafted features [20,69]. Mathematically, convolution is an efficient feature extraction approach [70], in which CNNs operate in parallel computing, allowing for the matrix operation to be sped up. The calculation of the output of the neural network is done using the following:

$$y = \sigma(\omega^L \cdots \sigma(\omega^2 \sigma(\omega^1 x + b^1) + b^2) \cdots + b^L), \tag{3.3}$$

where x is the input, ω is one set of network parameters, b is the bias, σ is the activation function, and y is the output neuron. The activation function plays an important role in DL to perform a multiple combination transformation. For example, the Rectified Linear Unit (ReLU) nonlinear activation function is defined as follows:

$$f(x) = \max(0, x), \tag{3.4}$$

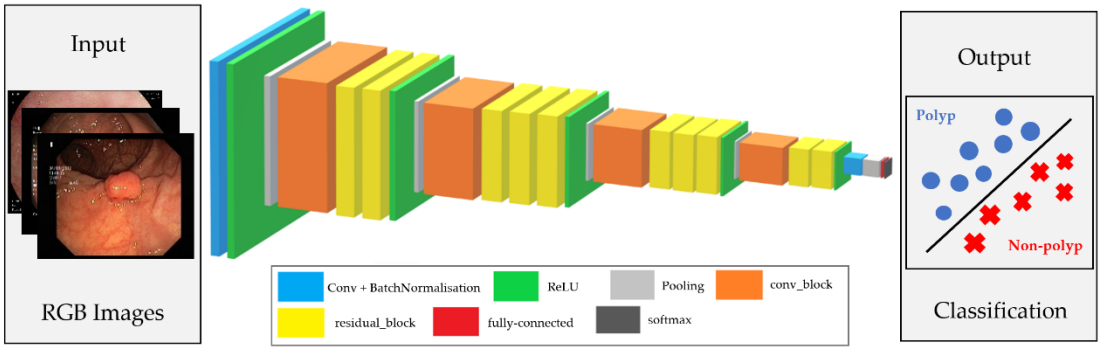


Figure 3.7: An illustration of the proposed ResNet-50 structure

Other than that, the Softmax function on the fully connected layer turns the logit scores into probabilities that equal 1. The equation is as follows:

$$S(y_i) = \frac{e^{y_i}}{\sum_{k=1}^{K} e^{y_k}}, for\ i = 1, \cdots, K, \tag{3.5}$$

where y is the input logit that takes the i^{th} vector value and K is the amount of real numbers for the probability distribution.

These extracted high-dimensionality features from CNN may cause the model overfitted because the model corresponds too closely to a particular set of data and does not generalize well. Hence, a dimensionality reduction technique is required to overcome the curse of dimensionality and avoid overfitting. Besides, different dimension reduction technique has different computational complexity and different implementation [71]. In order to find the most appropriate dimension reduction technique for our application, we tested our model with three different common dimension reduction techniques, PCA, t-Stochastic Neighbourhood Embedding (t-SNE), and Uniform Manifold Approximation and Projection (UMAP). In addition, PCA is faster than t-SNE and UMAP in terms of speed [72]. The results are shown in Table 3.4. In the experiment, it was shown that the PCA is more suitable in our application, with the best performance, followed by t-SNE, and UMAP. Therefore, PCA was adopted to reduce the dimensions of features extracted by CNN in our model.

Table 3.4: Performance comparison of the proposed model with different dimensionality reduction techniques

Techniques for dimensionality reduction	**Performance Evaluation Criteria**							
	MCC	**ACC (%)**	**SEN (%)**	**PRE (%)**	**SPEC (%)**	**TPR**	**FPR**	**AUC**
PCA (proposed)	0.9819	99.10	98.82	99.37	99.38	0.9882	0.0062	0.9995
t-SNE	0.7457	87.29	85.21	89.44	89.38	0.8521	0.1062	0.9271
UMAP	0.7278	86.33	83.75	88.67	88.98	0.8375	0.1102	0.9101

The feature representations from the training and testing images are obtained through the activations at the end of the networks, resulting in 2,048 extracted features in total. DL architecture is used to automatically create a feature vector before input to a classifier [73]. After the feature extraction by CNN, the feature set contains a higher dimension, which will lead to higher computation, and it consists of information redundancy [74]. With the feature reduction technique, the redundant features can be removed by speeding up the computation speed of the classifier while improving the accuracy of the classifier. Therefore, PCA reduces the dimensions of features that consist of many variations, while retaining the present variation (eigenvalues) in the dataset to the maximum extent before feeding into the classifier [75]. PCA is a statistical method that is widely used for data analysis and dimension reduction [76]. The idea behind PCA is to reduce the features to a lower-dimensional subspace to improve the performance of classifiers [74]. According to the eigenvectors in PCA, the original N-dimensional data are transformed to new M-dimensional data [77], in which the number of features is typically less than or equal to the number of original features; they will never be higher in number than the original ones. In this work, PCA explained 90% of the original variance, resulting in 227 features that could be described, meaning that the most significant features remain while the computation and information redundancy are kept at a minimum.

3.6 Ensemble Classification

Supervised learning is an ML technique; the goal of supervised ML is to learn a function based on labelled training data and make predictions based on testimony in uncertainty. Classifiers play an important role in distinguishing features in classification problems. In the past [78-80], various supervised ML algorithms have been used for classification tasks, such as SVM, Naïve Bayes, decision tree, nearest neighbour, ensemble methods, and so on. However, different classification algorithms may have different performances, depending on the application. Thus, a study was performed to compare the performance of different classifiers by testing them on our datasets. As a result, AdaBoost (M1) had the highest accuracy, followed by SVM (linear), K-Nearest Neighbour (5-nearest neighbour), Naïve Bayes (Gaussian) and

decision tree (CART). AdaBoost, short for adaptive boosting, is an iterative ensemble method that combines multiple poorly performing classifiers into a strong classifier for the classification problems instead of just an individual classifier which forms it. In general, AdaBoost often outperforms the single classifiers since it can learn non-linear decision boundaries [81]. Other than that, AdaBoost is amazingly fast since it takes less time to attain a similar learning accuracy [82] to other classifiers such as SVM. Therefore, AdaBoost was selected to execute the binary classification problem by distinguishing the polyp and non-polyp images in this study.

Adaboost was trained on the deep features extracted from the CNN after dimension reduction by PCA [74,83]. During each training iteration, the weight of each sample was altered based on the classifier error rate obtained [84], resulting in increasing or decreasing the weight of the misclassified data points by learning from previous errors. To train the AdaBoost classifier, 768 training data (input features) $T = \{(x_1, y_1), (x_2, y_2), \dots, (x_{768}, y_{768})\}$ were utilised; y_i is the output label and $y_i \in \{-1, +1\}$. The weight distribution of w_k^i is calculated over the training data 200 times (the maximum number of training iterations), as follows:

$$D_k = \left(w_1^i, w_2^i, \cdots, w_{200}^i\right), \tag{3.6}$$

with initialisation of $w_1^i = 1/n$, where $i = 1,2, \dots, 768$. After 200 iterations of training with weight distribution, D_k, the weak classifier, $f_k(x)$, is obtained, and a low weighted classification error rate (ε_k) for weak classifiers is computed relative to w_k^i in Equation (3.7).

$$\varepsilon_k = P(f_k(x_i) \neq y_i) = \sum_{i=1}^{n} w_k^i I(f_k(x_i) \neq y_i), \tag{3.7}$$

For the 200th weak classifier, $f_{200}(x)$, the weight coefficient is calculated as follows:

$$\alpha_k = 0.5 * \ln\left(\frac{1-\varepsilon_k}{\varepsilon_k}\right), \tag{3.8}$$

where the larger the error rate, ε_k, the smaller the corresponding weight coefficient, α_k, for a weak classifier. By way of explanation, a weak classifier with a small error rate has a larger weight coefficient [84]. To update the sample weight, D, the coefficient of the sample set is re-weighted, corresponding to the $(\mathrm{k}+1)^{\mathrm{th}}$ weak classifier:

$$w_{k+1}^i = \frac{w_k^i}{Z_k} * \exp\left(-\alpha_k y_i f_k(x_i)\right) = \begin{cases} \frac{w_k^i}{Z_k} * e^{\alpha_k} \; for \; f_k(x_i) \neq y_i \\ \frac{w_k^i}{Z_k} * e^{-\alpha_k} \; for \; f_k(x_i) = y_i \end{cases}, \tag{3.9}$$

$$Z_k = \sum_{i=1}^{200} w_k^i * \exp\left(-\alpha_k y_i f_k(x_i)\right), \tag{3.10}$$

where Z_k is a normalisation factor and is chosen so that D_{k+1} will be a probability distribution [82]. For example, the weight of the sample is increased in the $(k+1)^{\mathrm{th}}$ weak classifier if the classification is incorrect. In other words, the weight is decreased if the sample is correctly classified. Eventually, the final classifier, *Y(x)*, computes the sign of all weighted weak classifiers, as follows:

$$Y(x) = sign\left(\sum_{k=1}^{200} \alpha_k f_k(x)\right), \tag{3.11}$$

where the classifiers in the ensemble are aggregated into the final classifier, and this classifier is computed as a weighted majority vote of the weak classifiers (f_k). As there are more and more f_k (L increases), $Y(x)$ achieves a smaller and smaller error rate on the training data. Figure 3.8 demonstrates that the ensemble combines decisions from multiple classifiers into a final classifier $Y(x)$.

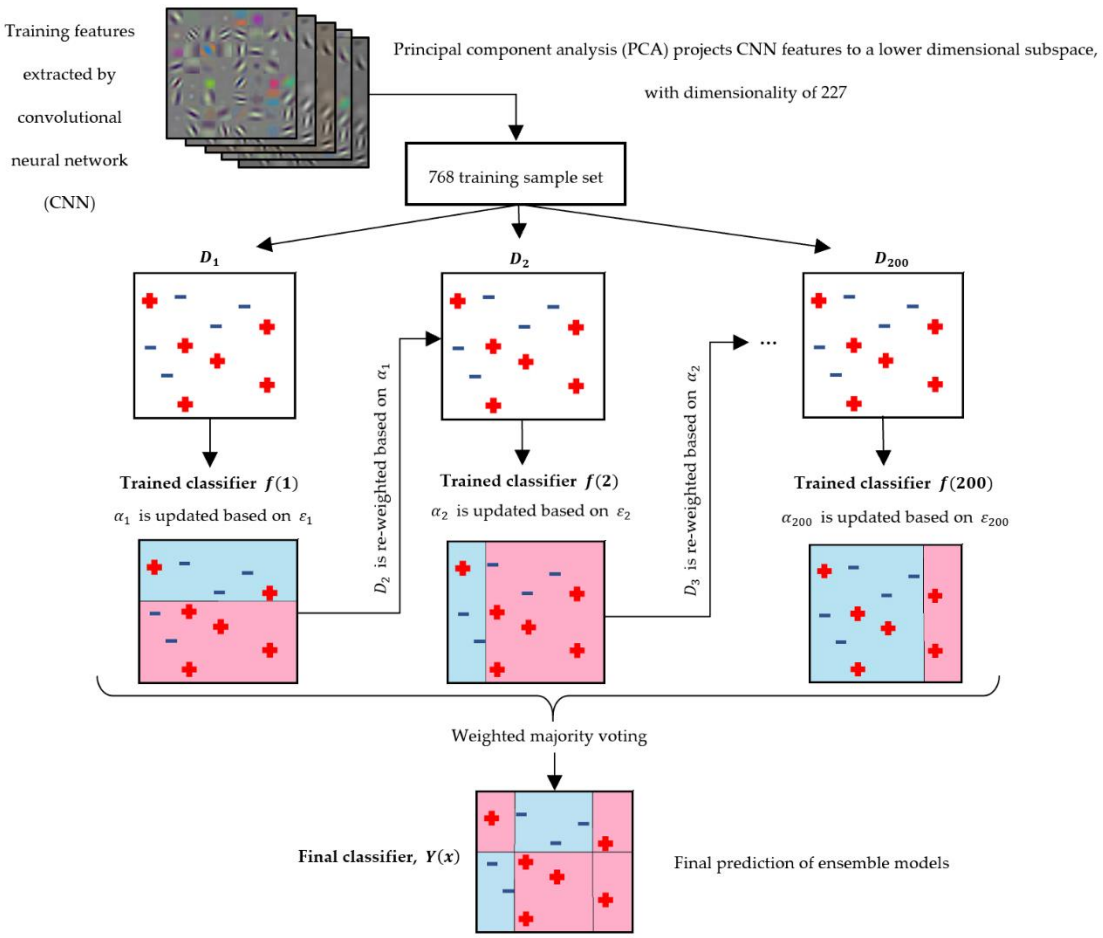


Figure 3.8: Illustration of the proposed methodology using ensemble learning (AdaBoost). The classifiers are trained iteratively based on the error made by previous models and the final prediction is based on the weighted majority vote among 200 classifiers

3.7 Performance Measure and Evaluation

A data analysis protocol (DAP) was adopted to the model to ensure reproducibility and prevent overfitting; the DAP following the guidelines derived by the MicroArray/Sequencing Quality control (MAQC/SEQC) initiatives led by the U.S. Food and Drug Administration [85-88]. The data were split into training and testing sets, the model undergoes a stratified 10 x 5-fold cross validation (CV). The performance of the model was evaluated in terms of the Matthews Correlation

Coefficient (MCC), accuracy, sensitivity, precision, and specificity. The evaluation metrics are defined as follows.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{3.12}$$

$$Accuracy\ (ACC) = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.13}$$

$$Sensitivity(SEN)/\ Recall = \frac{TP}{TP + FN} \tag{3.14}$$

$$Precision\ (PRE) = \frac{TP}{TP + FP} \tag{3.15}$$

$$Specificity\ (SPEC) = \frac{TN}{TN + FP} \tag{3.16}$$

where *TP*, *TN*, *FP*, and *FN* are obtained through a confusion matrix. (*TP*, true positive: a polyp is detected in a frame that contains a polyp; *TN*, true negative: no polyp is detected in a frame without a polyp image; *FP*, false positive: a polyp is detected in a frame without a polyp image; *FN*, false negative: a polyp is missed in a frame that contains a polyp). The MCC is a balanced measure of accuracy and precision, especially in binary classification, even when the classes are unbalanced [89]. Moreover, the AUROC curve was used to explore and visualise the model's performance. It is a graphical plot of the true positive rate (TPR) versus the false positive rate (FPR) at various thresholds. The TPR is also known as sensitivity/recall in Equation (3.14), and FPR is defined as follows:

$$FPR = \frac{FP}{FP + TN} \tag{3.17}$$

3.8 Hardware Implementation

In the hardware implementation of CNNs, Field Programmable Gate Arrays (FPGAs) have been often considered as an efficient platform for prototyping and performance exploration [90]. The FPGA exploits the inherent parallelism of CNNs and takes full advantage of its multiple multiply-accumulate units [91]. However, the FPGAs do not meet the design requirement of low-power consumption [44]. In addition, FPGAs suffer from a small internal memory [44].

A real-time application with low power consumption and high performance are the design specification for the hardware implementation of CNN. MSP430FR low-power microcontroller is thus used in this second phase of the research. MSP430FR series is a mixed-signal microprocessor, and it is designed in a 16-bit CPU with low power consumption. The microprocessor has a power consumption of 118µA/MHz in the active mode, less than 1µA in standby mode. In terms of intelligent digital peripheral, the microprocessor is a 32-bit hardware multiplier (MPY) with 6-channel internal direct memory access (DMA). In addition, the microprocessor has a ferroelectric random-access memory (FRAM) up to 256KB, which allows it has a fast and ultra-low-power writing speed at 125ns per word (64KB in 4ms). Apparently, the 32-bit hardware multiplier in multiply-and-accumulate (MAC) operation allows the convolutional layers to calculate the dot product efficiently. Other than that, the DMA feature allows several modules of the controller to direct access to the main system memory without computer intervention. Figure 3.9 below shows the MSP430FR5994 microprocessor on LaunchPad Development Kit. Figure 3.10 demonstrates the idea of prototype hardware for colonic polyp detection system which integrated SoC in the medical capsule robot, which consists of a battery supplied. Therefore, an ultra-low-power microprocessor MSP430FR series was utilised as a SoC and the CNN was implanted into the microprocessor to perform the colonic polyp detection task. In Figure 3.11, the overall bloack diagram is compared between the normal capsule robot and the proposed smart capsule robot. Nowadays, medical capsule robot was widely used for GI screening [92], and this is the idea that we can integrate the SoC with the camera of capsule endoscopy to help the physicians to detect the polyps automatically.

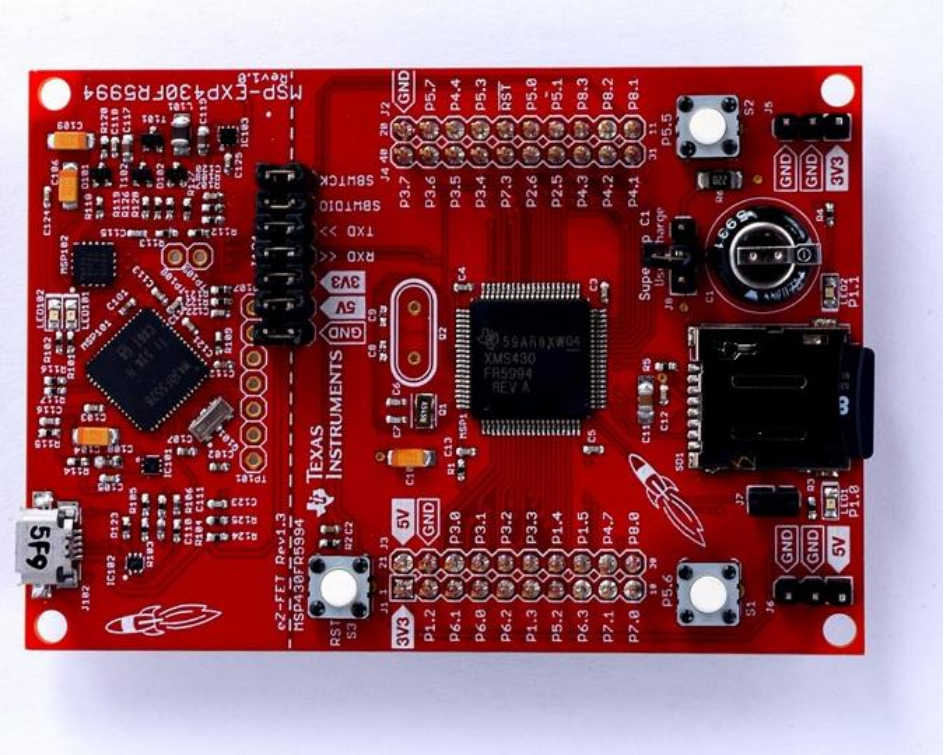


Figure 3.9: MSP430FR5994 LaunchPad Development Kit structure

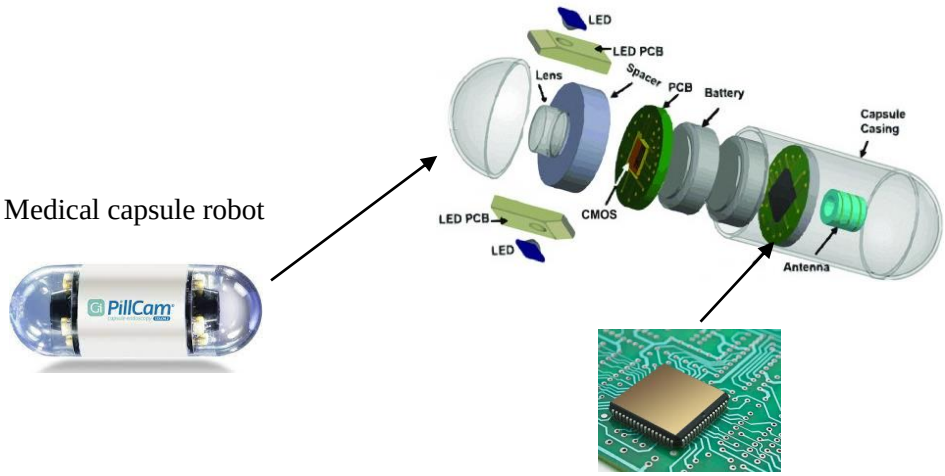


Figure 3.10: Idea-built prototype for the hardware implementation for the colonic polyp detection system

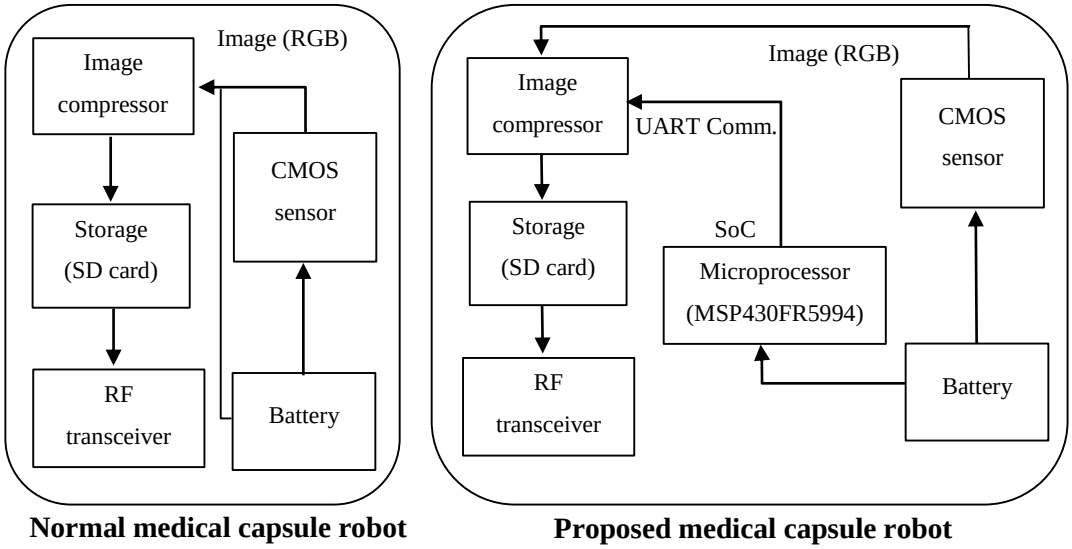


Figure 3.11: Comparison of block diagram between the normal medical capsule robot and proposed medical capsule robot

3.8.1 Network Design

Dataset 1,2, and 3 mentioned in Table 3.1 were merged to create training, validating, and testing datasets. In MATLAB, all images were normalized to 100 ×100 pixel and fed into the customized 4-layers neural network as shown in Figure 3.12. The designed network contains four layers with weights and biases (three convolutional layers and one fully-connected layer). The baseline CNN architecture was composed of three 2-D convolution layers with 8, 16, and 32 filters, respectively, and a filter size of 3 and stride of 1 for three layers. Zero padding was applied on the first convolution layer to prevent the decimal activation on the latter operations. Furthermore, every 2-D convolution layer was followed by ReLU activation function and max-pooling layer that performed downsampling with a filter size of 2 and stride of 2. Finally, a fully-connected layer was applied as a classifier to obtain the predicted class as output, while a softmax activation function was utilised by the output layer to convert the output from the fully-connected layer into a probability distribution. The model was optimised using the binary cross-entropy loss function and the SGDM.

After training the model with 30 epochs and a mini-batch size of 128, an accuracy validation using the validation dataset was performed. As shown in Figure 3.13, the

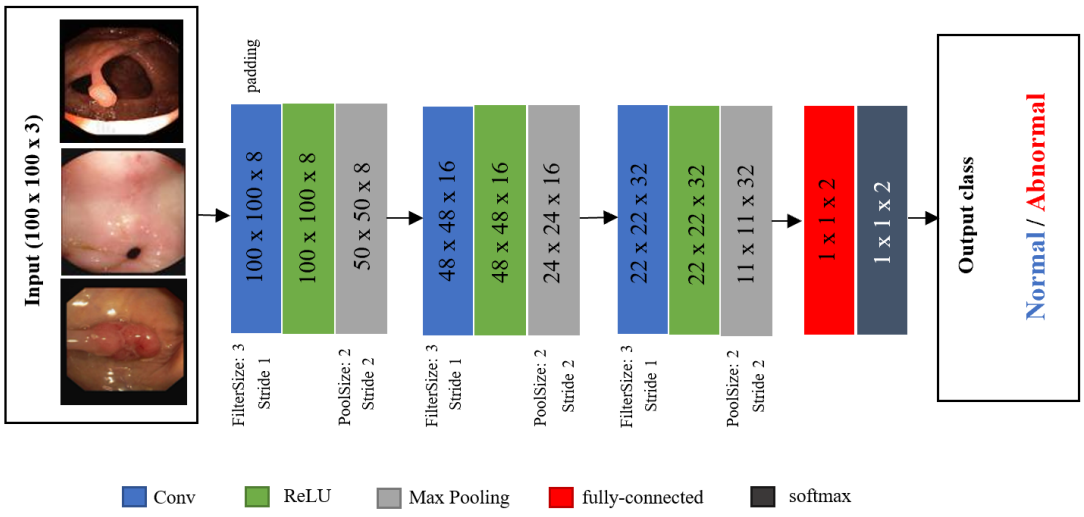


Figure 3.12: Proposed 4-layers neural network

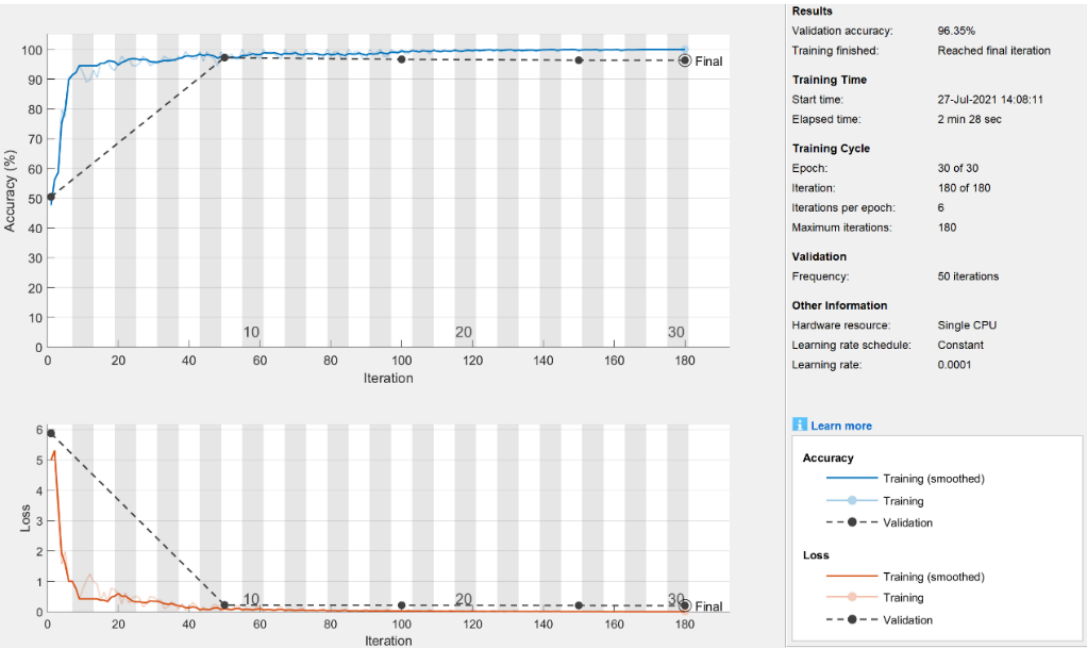


Figure 3.13: Training process for the 4-layers network model

validation accuracy obtained for the proposed 4-layers network was equal to 96.35%. In Table 3.5, the 4-layer model was compared with other well-known pre-trained deep neural network models. Different number of convolutional layers and the number of filters per layer, plus the size of the filters and the type of downsampling operation could affect and improve the accuracy. Due to the memory constraint on the microprocessor, the 4-layers network is proposed to be implanted on the microprocessor, instead of choosing the pre-trained networks, which consist of more deeper layers.

Table 3.5: Architecture and accuracy of CNN models

CNN model	**Number of layers**	**Number of conv. filters**	**Size of conv. filters**	**Size of pool. filters**	**Accuracy (%)**
AlexNet	8	96-256-384-384-256	11-5-3-3-3	3-3-3	51.37
GoogLeNet	22	64-64-192-(64-96-128-16-32-32)-(128-128-192-32-96-64)-(192-96-208-16-48-64)-(160-112-224-24-64-64)-(128-128-256-24-64-64)-(112-144-288-32-64-64)-(256-160-320-32-128-128)-(256-160-320-32-128-128)-(384-192-384-48-128-128)	7-1-3-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)-(1-1-3-1-5-1)	3-3-3-3-3-3-3-3-3-3-3-3-3	94.68
ResNet-50	50	64-(64-64-256-256)-(64-64-256)-(64-64-256)-(128-128-512-512)-(128-128-512)-(128-128-512)-(128-128-512)-(256-256-1024-1024)-(256-256-1024)-(256-256-1024)-(256-256-1024)-(256-256-1024)-(256-256-1024)-(512-512-2048-2048)-(512-512-2048)-(512-512-2048)	7-(1-3-1-1)-(1-3-1)-(1-3-1)-(1-3-1-1)-(1-3-1)-(1-3-1)-(1-3-1)-(1-3-1-1)-(1-3-1)-(1-3-1)-(1-3-1)-(1-3-1)-(1-3-1-1)-(1-3-1)-(1-3-1)	3	96.94

Table 3.5: Architecture and accuracy of CNN models (Cont.)

CNN model	**Number of layers**	**Number of conv. filters**	**Size of conv. filters**	**Size of pool. filters**	**Accuracy (%)**
VGG-19	19	64-64-128-128-256-256-256-256-512-512-512-512-512-512-512-512	3-3-3-3-3-3-3-3-3-3-3-3-3-3-3-3	2-2-2-2-2	96.28
Proposed (4-layers network)	4	8-16-32	3-3-3	2-2-2	94.53

3.8.2 Development of Basic Funcitions of CNN on microprocessor

To assess the practicality of the proposed idea, the basic functions of the CNN are implemented on the microprocessor to prove the concept of implantation of SoC into the medical capsule robot. The basic functions such as padding, convolution, ReLU, max-pooling, fully-connected, and softmax were developed and debugged through an IDE. The IDE used to develop those CNN's functions for the embedded processor is the CCS version 10 from Texas Instruments. Figure 3.14 shows the user interface of the CCS.

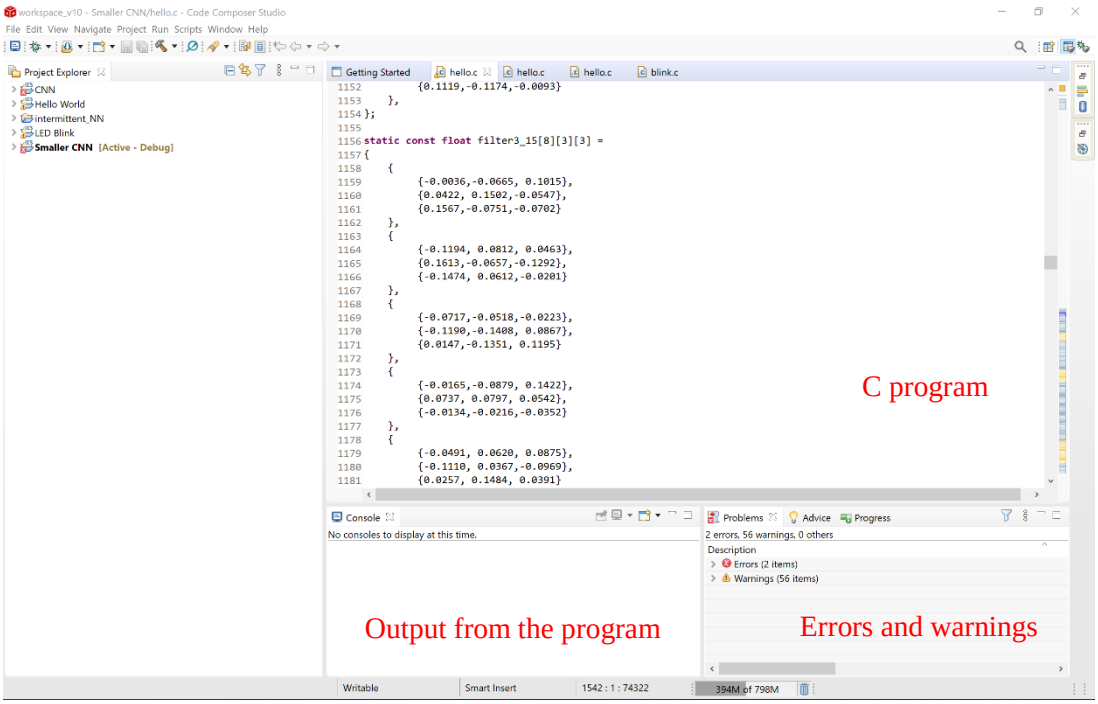
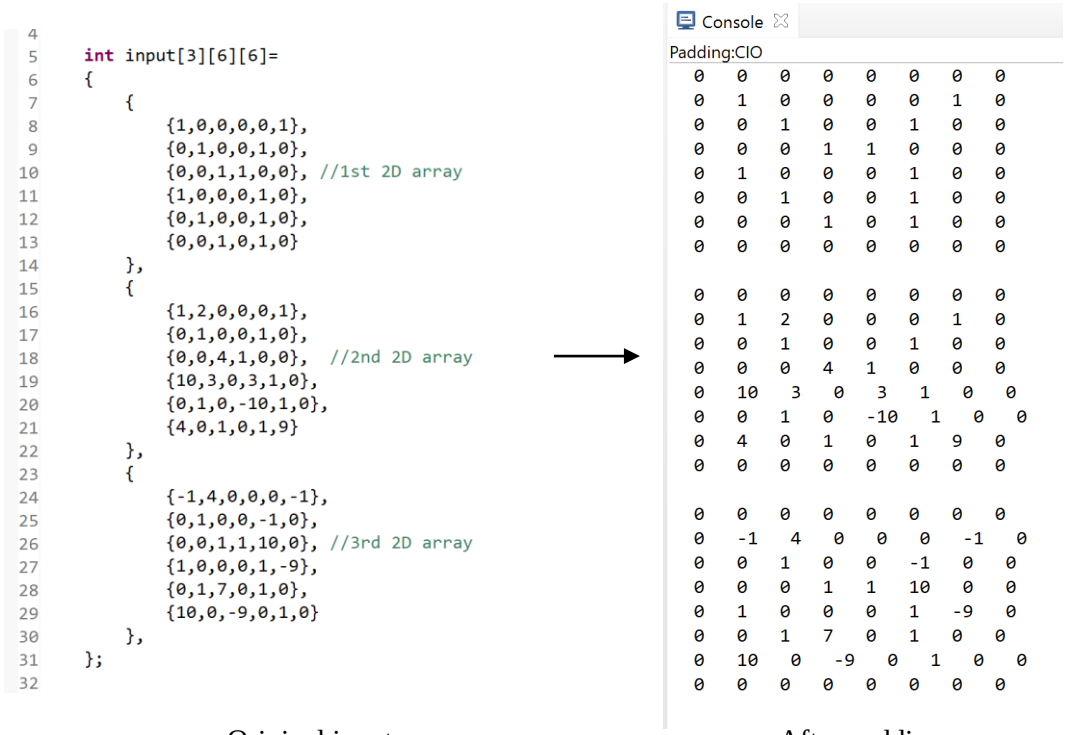


Figure 3.14: User interface of CCS used for the simulation

3.8.2.1 Zero Padding

Due to the decimal output after the pooling operation on the odd numbers, the padding operation is applied before the convolution operation. In zero padding, a border of pixels with zero value was added around the edges of the input. The purpose of using this zero padding technique is to preserve and maintain the original size/dimensions of input after the convolutional layer. For instance, a colour image has

3 colour channels, a random 6 × 6 × 3 input is thus used as an input. As shown in Figure 3.15, the output after padding will become 8 × 8 × 3.



Original input After padding

Figure 3.15: Output after zero padding operation

3.8.2.2 Convolution

In the convolutional layer, the output after zero padding will undergo the sliding window operation with the filters by producing new feature maps. During the sliding convolutional, the layer convolves the input by moving the filters along the input horizontally and vertically by calculating the dot product of the weights and the input and lastly adding a bias term. Those filters and bias are network parameters to be learned. Besides, the output of the feature maps after convolution operation depends on the parameters such as the size of the filter, sliding window step size (stride), and the number of filters. In Figure 3.16, the convolutional layer applied sliding convolutional filters to the input (8 × 8 × 3), which is the output from padding. The parameters of the convolution layer were set as 3 × 3 filter size, stride of 1, and 2 filters.

```
int filter1[3][3][3]=
 {
    {
         {1,-1,1},
         {1,0,-1}, //1st 2D array
         {1,0,0}
     },
     {
         {-1,0,-1},
         {0,-1,0},  //2nd 2D array
         {0,0,0}
     },
     {
         {-1,-1,-1},
         {-1,1,-1},  //3rd 2D array
         {0,-1,1}
     },
 };

 int filter2[3][3][3]=
 {
    {
         {1,-1,1},
         {1,0,-1}, //1st 2D array
         {1,0,0}
     },
     {
         {-1,0,-1},
         {0,-1,0},  //2nd 2D array
         {0,0,0}
     },
     {
         {-1,-1,-1},
         {-1,1,-1},  //3rd 2D array
         {0,-1,1}
     },
 };

int bias1 = 2;
int bias2 = 10;
```

```
Console
CNN function:CIO
First feature map:
  -3   5   -1   1   3   1
  -6   0   -4   12   -8   4
  0   -1   -4   -6   2   3
  -6   1   -7   -18   -1   -16
  -15   -24   12   4   5   11
  7   -9   -6   2   10   -8

Second feature map:
  5   13   7   9   11   9
  2   8   4   20   0   12
  8   7   4   2   10   11
  2   9   1   -10   7   -8
  -7   -16   20   12   13   19
  15   -1   2   10   18   0
```

Figure 3.16: New feature maps generated in the convolutional layer

3.8.2.3 ReLU

To prevent the vanishing gradient problem, the ReLU activation function improves efficient computation by allowing the models to learn faster and perform better. The function returns all the negative input values to zero, while the positive values remain those values back. Figure 3.17 demonstrates the feature maps that undergo ReLU activation.

```
Console
CNN function:CIO
First feature map after ReLU activation:
  0   5   0   1   3   1
  0   0   0   12   0   4
  0   0   0   0   2   3
  0   1   0   0   0   0
  0   0   12   4   5   11
  7   0   0   2   10   0

Second feature map after ReLU activation:
  5   13   7   9   11   9
  2   8   4   20   0   12
  8   7   4   2   10   11
  2   9   1   0   7   0
  0   0   20   12   13   19
  15   0   2   10   18   0
```

Figure 3.17: Output maps after ReLU activation

3.8.2.4 Max-pooling

To reduce the dimensions of the feature maps and summarise the features in a region of the feature map produced by a convolutional layer, the max-pooling operation is utilized to select the maximum element from the region of the feature map covered by the filter. Therefore, the feature map after the max-pooling layer contains the most prominent features of the previous feature map. Furthermore, there are some parameters for the max-pooling operation (i.e., pool size and stride). Different parameters will produce different output. For example, in Figure 3.18, the feature maps (6 × 6) after the ReLU function were subsampling into 3 × 3 in the condition of stride of 2 and pool size of 2 × 2.

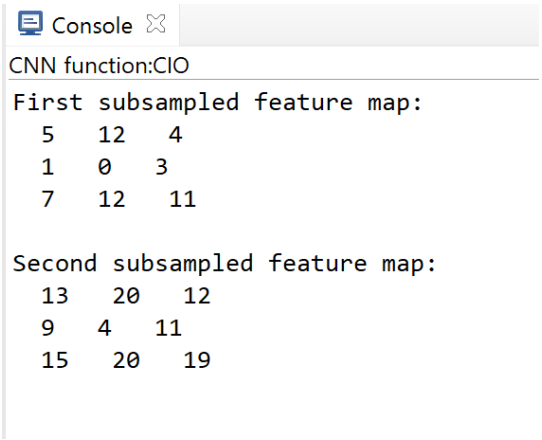


Figure 3.18: Subsampled feature maps (output) after the max-pooling layer

3.8.2.5 Fully-connected

Before the fully-connected layer, the output from the final pooling or convolutional layer was flattened into a vector. The flattened vector was connected to a few fully-connected layers. In the fully-connected layer, the dot product of the weights and flattened vector was computed, and lastly, adding a bias term. In our case, we have only two classification outputs, which are polyp (abnormal) and non-polyp (normal) for the endoscopy images. Therefore, there are only two fully-connected layers. Figure 3.19 shows the calculation and result for two fully-connected layers.

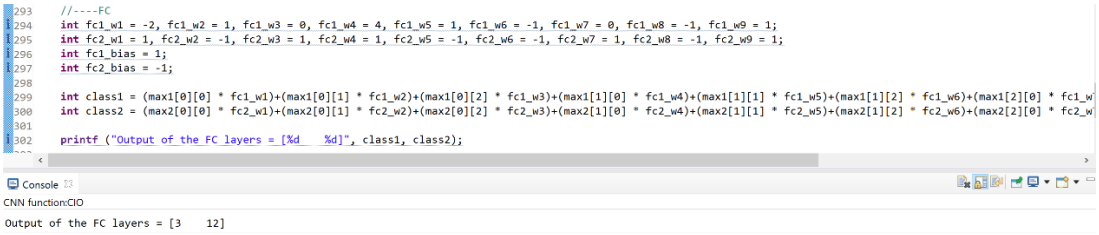


Figure 3.19: Output vectors of the fully-connected layers

3.8.2.6 Softmax

In the final layer, the softmax activation function was used to obtain the probabilities of the input being in a particular class. The probability values were generated to the respective classes after the activation function, for softmax, the sum of the probabilities is equal to 1. Based on the probability values of each class, we can know that the input (endoscopy image) belongs to polyp or non-polyp. As shown in Figure 3.20, it was clear that the input belongs to class 2 (image without polyp).

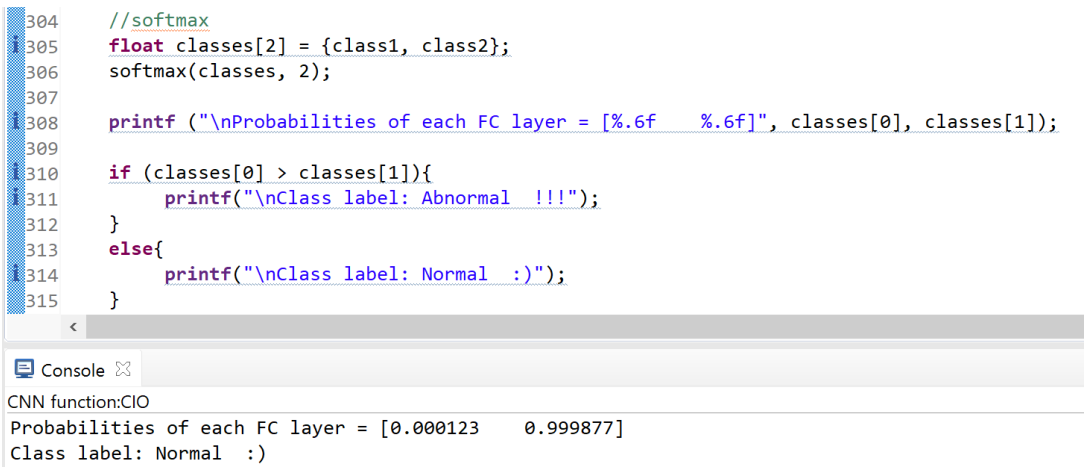


Figure 3.20: Probability values generated for each class after the softmax activation function

3.9 Summary

Based on the aforementioned studies and the research assumptions, a comprehensive examination of each phase provides the most appropriate method in achieving a

reliable diagnosis system. The DL and ML techniques were integrated into the proposed CAD system for colonic polyp detection (phase I algorithm). Besides, for hardware implementation, the DL technique was utilised into the proposed smart medical capsule robot (phase II algorithm).

In phase I (software implementation), a modified ResNet-50 was used as a feature extractor, with PCA and ensemble learning classifier (AdaBoost) to classify the colonic polyp images. The architecture of ResNet-50 was altered in order to reduce the computational cost, while retain or improve its performance for the application of colonic polyp detection. For the network training, in Table 3.3, the modified ResNet-50 obtained a better accuracy, with a lower computational time. Besides, it has a lower network complexity as compared with the original ResNet-50. Thus, the modified ResNet-50 was used as a feature extractor to extract the features of images. Next, three dimension reduction techniques (PCA, t-SNE, and UMAP) was tested on the model in order to find the most appropriate dimension reduction technique for our application. In Table 3.4, it was shown that the PCA is the most suitable dimension reduction technique for our application. Lastly, the ensemble classifier, AdaBoost, is selected as the classifier of the model. This is because a study was performed to compare the performance of different classifiers by testing them on our datasets. As a result, AdaBoost (M1) had the highest accuracy, followed by SVM (linear), K-Nearest Neighbour (5-nearest neighbour), Naïve Bayes (Gaussian) and decision tree (CART). Based on the aforementioned studies, the best and appropriate feature extractor, dimension reductor, and classifier were chosen and combined as the classification model for the software implementation. The proposed model was analysed based on six evaluation methods, which are the MCC, accuracy, sensitivity, precision, specificity, and FPR.

The different implementation between phase I and phase II algorithms is the CNN network utilised as a feature extractor in phase I; meanwhile, the CNN was used as a classifier in phase II. Besides, the CNN network used in phase I is different with the CNN used in phase II. This is because the modified ResNet-50 is not able to fit into the microprocessor due to its memory constraint. Thus, a 4-layer network was proposed in phase II for the hardware implementation of CNN on the medical capsule robot.

CHAPTER 4

RESULTS AND DISCUSSION

The results obtained for the proposed method in both phases were discussed in this chapter. A thorough analysis was carried out on both phases to validate the effectiveness of the suggested approach for the CAD system and hardware implementation of CNN on the capsule robot.

4.1 Performance of Computer-Aided Diagnosis (CAD) System

The performances of the classification models are reported in Table 4.1, showing that our proposed method using the modified ResNet-50 with a smaller architecture size obtained increasing gains in every metric, except sensitivity, as compared to the original ResNet-50. In Table 4.1, the area under the curve (AUC) is a divisibility measurement that measures the ability of the model to distinguish between different classes. A higher value of AUC indicates better performance of the model. However, in our proposed algorithm, the modified version of ResNet-50 had a slightly higher AUC than the original ResNet-50. The representative receiver operating characteristic (ROC) curves for different approaches are displayed in Figure 4.1, based on Table 4.1.

This study proposed a new combination of modified deep residual convolutional neural networks (ResNet-50) with the PCA and ensemble learning (AdaBoost) approach for a colonic polyp classification system. Compared to the original ResNet-50, the modified ResNet-50-meta architecture achieved state-of-the-art results. Concurrently, it had a smaller size with a lower complexity of the architecture. As a result, the performance of the model was improved while reducing its computational time. Additionally, an adaptive boosting-based ensemble classifier was trained on the principal component of feature extraction with the class labels (non-polyp or polyp) from the training dataset. In the trial of 1,517 images from a combination of three free,

Table 4.1: Performance comparison between the modified ResNet-50 and the original ResNet-50

Method	Performance Evaluation Criteria							
	MCC	**ACC (%)**	**SEN (%)**	**PRE (%)**	**SPEC (%)**	**TPR**	**FPR**	**AUC**
ResNet-50	0.9270	96.35	96.45	96.26	96.25	0.9645	0.0375	0.9935
Original ResNet-50 as feature extractor with PCA and AdaBoost	0.9754	98.77	99.41	98.14	98.12	0.9941	0.0188	0.9994
Modified ResNet-50 as feature extractor with PCA and AdaBoost	0.9819	99.10	98.82	99.37	99.38	0.9882	0.0062	0.9995

publicly accessible databases, the proposed algorithm obtained good results with 0.9819 MCC. The accuracy, sensitivity, precision, specificity, TPR, FPR, and AUC of polyp classification were 99.10%, 98.82%, 99.37%, 99.38%, 0.9882, 0.0062, and 0.9995, respectively. Hence, these results show that the proposed algorithm is robust enough to assist in CAD.

Additionally, an experiment was done with Table 4.2 using only Dataset 1 (Kvasir) with our proposed algorithm (modified ResNet-50 with AdaBoost). This is because

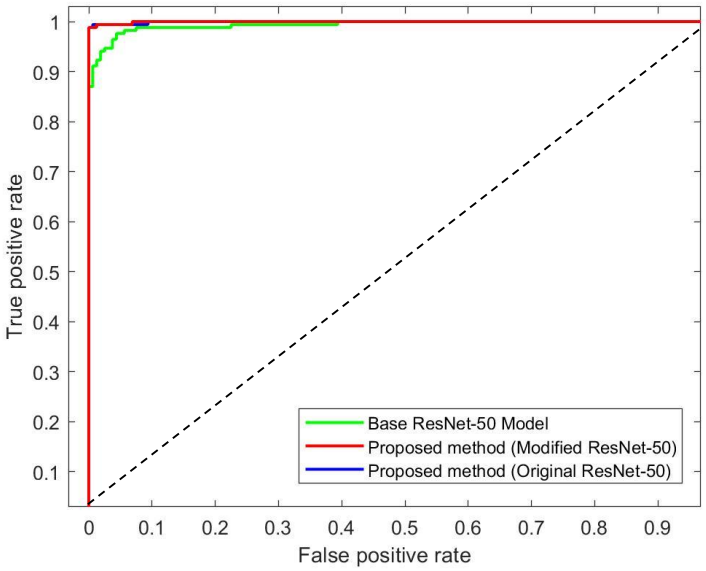
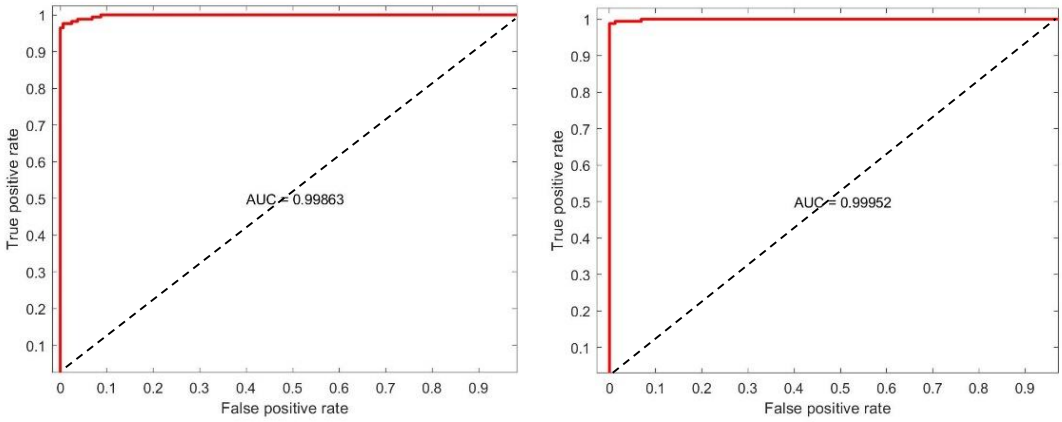


Figure 4.1: Receiver operating characteristic (ROC) graph for different architectures of ResNet-50

Table 4.2: Performance comparison between the modified ResNet-50 and the original ResNet-50

Database	**Performance Evaluation Criteria**							
	MCC	**ACC (%)**	**SEN (%)**	**PRE (%)**	**SPEC (%)**	**TPR**	**FPR**	**AUC**
Dataset 1	0.9586	97.91	96.45	99.35	99.38	0.9645	0.0062	0.9986
Dataset 2	N/A	N/A	20.18	N/A	N/A	N/A	N/A	N/A
Dataset 3	N/A	N/A	100	N/A	N/A	N/A	N/A	N/A
Dataset 1, 2, and 3	0.9819	99.10	98.82	99.37	99.38	0.9882	0.0062	0.9995

only Dataset 1 consists of two classes of data. Seventy percent of images were selected randomly for training; the remaining 30% were used for testing. Similarly, the same model was tested on Datasets 2 and 3 (ETIS-LaribPolypDB and CVC-ClinicDB). Due to the unavailability of non-polyp images (normal colon) in both datasets, FP and TN were zero. The representative ROC curves for Table 4.2 are displayed in Figure 4.2.



(a) **(b)**

Figure 4.2: Receiver operating characteristic (ROC) graph for (a) Kvasir (Dataset 1) database only; (b) a combination of three databases

As we can see, the combination of three datasets helped increase the performance of the model instead of just using Dataset 1 to train the CAD model. The purpose of using different datasets is because we want to test how robust our model is. Therefore, the most accurate model for the CAD system was obtained by combining three databases that consist of more features learned from Dataset 2 and 3. For this model, the MCC, accuracy, sensitivity, precision, and specificity of the endoscopic image's classification task were 0.9819, 99.10%, 98.82%, 99.37%, and 99.38%, respectively. Table 4.3 compares the previous method with our proposed method; our method achieved a better performance than all the other methods. For instance, Nadimi et al. [40] had the best results among the comparison. As compared with them, our detection model was increased in terms of 1.1% accuracy, 0.72% sensitivity, and 3.08% specificity. Another comparison was performed against Wittenberg et al. [35]; among their datasets are two datasets that are similar to our dataset. In comparison with their work, the performance of our model was increased in terms of 11.82% sensitivity. Those N/A values for the previous works in Table 4.3 were unable to compute as they did not provide the confusion matrix in their research, such as the TP, TN, FP, and FN. Some figures of our experimentation result are shown in Table 4.4.

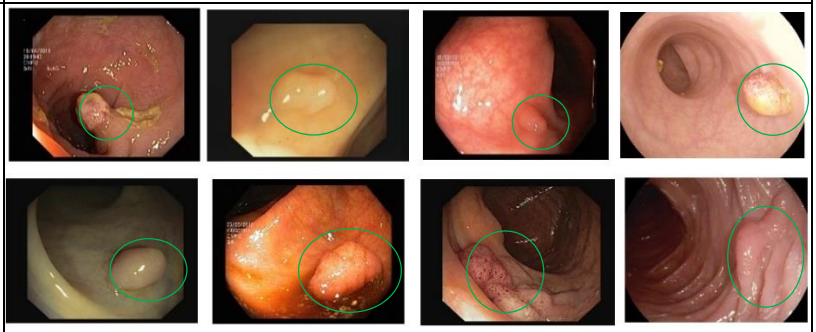
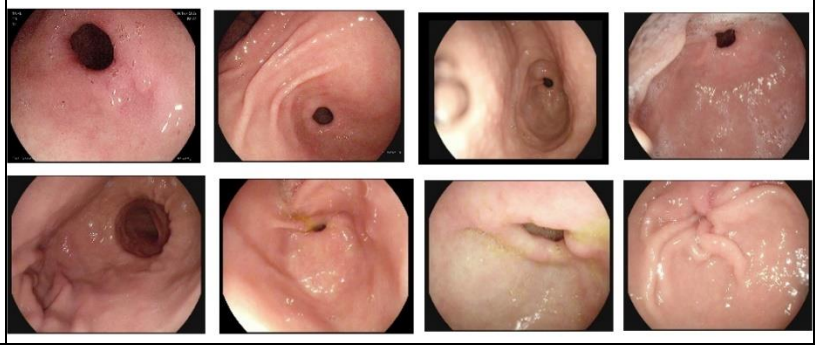
Table 4.3: Performance comparison between previous works and our proposed method

Method	Database	Technique	Performance Evaluation Criteria				
			MCC	ACC	SEN	SPEC	FPR
Proposed	Dataset 1, 2, and 3	• Preprocessing • Modified ResNet-50 (FE) • PCA • Ensemble (AdaBoost)	0.9819	99.10	98.82	99.38	0.0062
Wittenberg et al., (2019) [35]	Dataset 2, 3, and "Bayreuth"DB	• Mask R-CNN • ResNet-101 (FE) • TFL	N/A	N/A	87.00	N/A	N/A
Liu et al., (2019) [37]	Dataset 2, 3, and CVC-ColonDB	• Preprocessing • SSD framework • InceptionV3 (FE)	N/A	N/A	80.30	N/A	N/A
Vani et al., (2019) [39]	CVC-ColonDB, WCE video frames and endoscopy images from Endoatlas and Shaily	• Data augmentation • DL technique (VGG-19) using Keras framework	N/A	94.45	94.00	N/A	N/A

Table 4.3: Performance comparison between previous works and our proposed methods (Cont.)

Method	Database	Technique	Performance Evaluation Criteria				
			MCC	ACC	SEN	SPEC	FPR
Nadimi et al., (2020) [40]	Images captured from colon capsule endoscopy	• Data augmentation • CNN TFL (ZF-Net) • SGDM	N/A	98.00	98.10	96.30	N/A
Patino-Barrientos et al., (2020) [3]	Private dataset from University of Deusto	• Preprocessing • 4 layers base CNN model • VGG-16 (FE)	N/A	83.00	86.00	N/A	N/A

Table 4.4: Example of images that classified correctly during the real experiment

Actual Class	**Predicted Class**	**Images**
Polyp	Polyp	
Non-polyp	Non-polyp	

Moreover, the performance of the model was estimated through the plot of the generalisation error values during the learning process in Figure 4.3. The generalisation error and overfitting are closely related since the generalisation error is computed to measure the ability of the model to predict the outcome values for the formerly unseen data. The smaller the generalisation error, the less overfitting that occurred. According to Figure 4.3, the cumulative generalisation error decreased to approximately 2% when 150 weak learners composed the ensemble classifier.

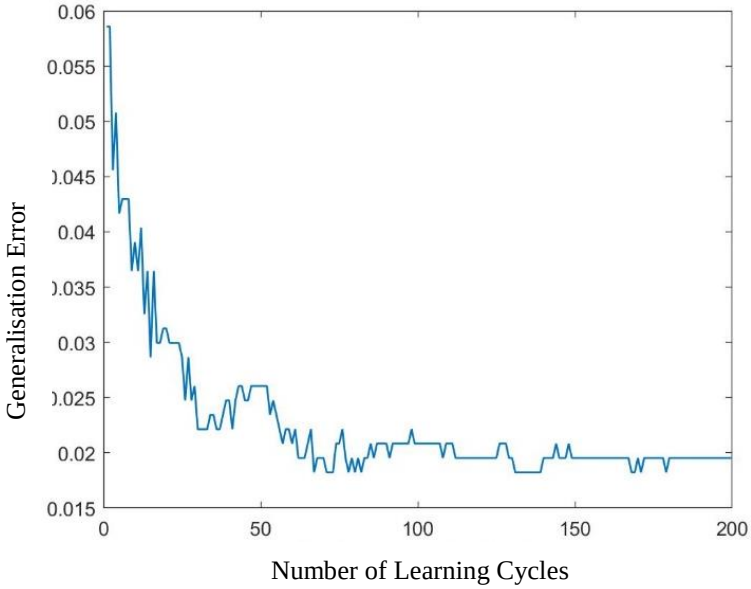


Figure 4.3: Generalisation error versus number of learning cycles

4.2 Datasets and Misjudgement by the CAD System

For better performance of the model, three datasets were merged to train and validate the model. Nevertheless, within the datasets, there were slightly more images with than without a polyp. To prove the proposed method is a stable algorithm as there is no subjective bias in polyp recognition with balanced and imbalanced datasets. Therefore, an analysis was performed, as shown in Table 4.5, indicating that the model was not biased towards the majority class. The two different classes (polyp and non-polyp) were separated into three different ratio configurations, where i) the number of images with a polyp was equal to those without a polyp; ii) there were 30% more images with a polyp than without a polyp, and iii) there were 50% fewer images with a polyp than without a polyp, which corresponds to the common distribution in the real world. On the balanced dataset, the classification model achieved a MCC of 0.9788, with accuracy, sensitivity, precision, and specificity of 98.96%, 98.02%, 98.88%, and 99.06%, respectively. Meanwhile, on the imbalanced dataset, the model obtained a MCC of 0.9103, with accuracy, sensitivity, precision, and specificity of 95.67%, 97.81%, 96.12%, and 96.83%, respectively, when the number of images with a polyp was 30% more than the number without. Contrarily, when the number of images with a polyp was 50% less than the number without a polyp, the MCC, accuracy, sensitivity,

Table 4.5: Performance of the proposed model, with balanced and imbalanced datasets. P - polyp, NP - non-polyp

Evaluation of the model	**Number of images**		
	Balanced dataset	**Imbalanced dataset**	
	P: 532; NP: 532	**P: 565; NP: 396**	**P: 266; NP: 532**
MCC	0.9788	0.9103	0.8549
ACC (%)	98.96	95.67	92.10
SEN (%)	98.02	97.81	91.99
PRE (%)	98.88	96.12	92.33
SPEC (%)	99.06	96.83	93.09

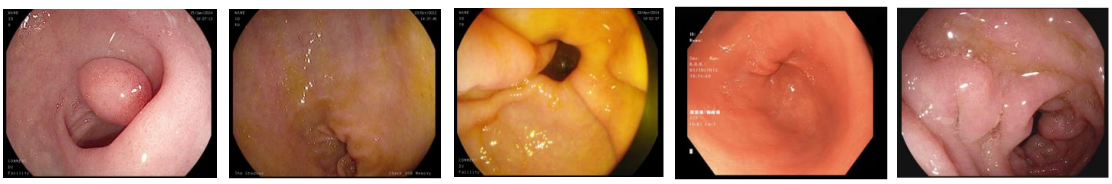
precision, and specificity of the classification model were 0.8549, 92.10%, 91.99%, 92.33%, and 93.09%, respectively.

From the combination of three datasets, various features are acquired in the proposed detection model, including the histologic features (i.e., tubular, villous, and tubulovillous) and morphologic features (i.e., colour, shape, texture, sessile, pedunculated, and flat). The detection model achieved a good performance in detecting the polyps by extracting these essential pathological features of colorectal lesions [93,94]. However, for radiomic, the features include spatial features and morphologic features; those features are computed based on the statistical descriptors [95,96]. For spatial features, the features were extracted based on the gray-scale level, which consists of first-order statistics features and gray level co-occurrence matrix texture features [96]. Not only that, but there are also other feature classes, i.e., gray level difference method, gray level size zone matrix, gray level run length matrix, and neighbourhood gray-tone difference matrix [97]. For morphologic features, the features were detected through the shape and the area of the lesion [96,97].

However, the automated colonic polyp classification model still has limitation as it suffered from inaccuracy or misclassification of endoscopic images. It was found that the problem that led to the misjudgment of images was due to very similar characteristics between polyp and non-polyp images, such as the texture, density,

colours, and shape information of the colon, causing the system to misjudge irrelevant objects that resemble polyps. Hence, it was hard to discriminate the features between them. Figure 4.4 shows the images that were misclassified in our polyp classification model. In Figure 4.4a, the image was classified as a non-polyp image. In contrast, the images in Figure 4.4b-e were misclassified as polyps detected.

In Figure 4.4a, the polyp was incorrectly detected due to its context. This is because the image has a similar colour feature as non-polyp images, and the polyp was separated as background during the pre-processing steps. In Figure 4.4b, the non-polyp image was classified as a polyp image, since the structure of the colon resembles polyps. Additionally, the colour of that image is darker, which means it has similar colour features as polyp images. In Figure 4.4c, the image is misclassified as a polyp because of the natural convex structure of the colon, which made the image have features similar to a polyp's shape. Figure 4.4d was misjudged as a polyp image because the colour intensity in the image was identical to that of an inflamed polyp. For Figure 4.4e, the image was identified as a polyp image due to the irregular texture of the colon that resembling a bulging polyp shape. Moreover, the colour properties and density of the images taken by endoscopy also one of the cause that led to the misjudgment of images. This limitation can be improved in the future by using more advanced pre-processing techniques if it is to be utilised in colonoscopy for real-time



Actual class: Polyp	Actual class: Non-polyp	Actual class: Non-polyp	Actual class: Non-polyp	Actual class: Non-polyp
Predicted class: Non-polyp	Predicted class: Polyp	Predicted class: Polyp	Predicted class: Polyp	Predicted class: Polyp
(a)	**(b)**	**(c)**	**(d)**	**(e)**

Figure 4.4: Misclassified images. **(a)** Image misjudged as non-polyp. **(b)-(e)** Images misjudged as polyps

detection. Therefore, pre-processing steps are very important to discriminate the features of the polyp and non-polyp images.

In addition, though CNNs have shown promise in the past few years, the reported accuracy, sensitivities, and specificities of the CNNs in the literature vary distinctly [98]. At present, there are many exaggerated claims that there is a risk for patient safety and population health, with the AI algorithms practiced in some cases to millions of patients [99]. It is causing the provision of improper care that is not in the best interests of the patients. In the future, the proposed CAD system will be evaluated by conducting an appraisal of the methods, adherence to reporting standards, risks of bias, and claims of DL studies that compare diagnostic AI performance with human physicians [99]. For example, we can establish randomised clinical trials by undergoes the assessment with or without an AI platform diagnosis. Thus, validation of reported results in a large trial is required to enhance the effectiveness of CNN systems [98,100]. By doing so, higher quality and more distinctly reported evidence base could help to lessen research waste and protect patients.

4.3 Hardware Implementation of CNN on the Microprocessor

In the second phase of the research, a CNN-based model was developed on the microprocessor (SoC) and integrated into the medical capsule robot. Nevertheless, the difficulty with this approach remains the constraint of implementing CNN layers in the hardware due to the network's complexity and the memory required. Therefore, a 4-layers network is designed and trained through MATLAB using Dataset 1, 2, and 3. From the MATLAB, the designed network obtained a training accuracy of 96.35%, and during the testing, an accuracy of 92.86% is obtained. All the network's parameters (i.e., weights and biases) are then reproduced to the functions developed in CCS for the microprocessor.

For the hardware implementation, a medical capsule robot with a battery supplied, the choice of the best colonic polyp detection model does not only depend on its performance, yet the power consumption and memory footprint must be considered. Due to the limited FRAM in the microprocessor, the complete 4-layers network is not

able to flash to the microprocessor. To reduce the memory usage, some of the local variables have been moved to global variables. The variables for the network's parameters have been changed to constants, but there is still the same issue. Therefore, in Table 4.6, all the developed functions are combined and tested with small input (6 × 6 × 3), small filter/pool size (3 × 3/2 × 2), and small number of filters to compute the cumulative memory (FRAM and SRAM) used in each operation, as well as the number of neurons. The FRAM and SRAM refer to Ferroelectric Random-Access Memory and Static Random-Access Memory used in the microprocessor.

Furthermore, in Table 4.7, the memory required in each layer is computed, based on the proposed 4-layers network, as well as the floating point operations (FLOPs), to determine the computational complexity of the network. For the writing and reading speed, it is limited to 8 MHz as the FRAM limits the maximum clock speed of the microcontroller. The controller can run higher clock speeds, but the increasing CPU clock speed would cause additional wait states that lower the power efficiency of the controller. FRAM allows for the continuous ultra-low power data logging, which bring cost, energy and efficiency optimization. In this way, the hardware implementation that used as an embedded device for real-time applications can achieved a low power consumption.

Table 4.6: Number of neurons and memory required over each network layer

Layer	**Operation**	**Number of neurons**	**Cumulative memory (byte)**	
			FRAM	**SRAM**
1	Input (6 × 6 × 3)	-	-	-
2	Padding (8 × 8 × 3)	-	19,370	1,340
3	2 × Conv2D (3 × 3)	72	20,180	1,604
4	ReLU	-	20,706	1,604
5	2 × MaxPool2D (2 × 2)	18	22,276	1,716
6	FC	2	22,582	1,716
7	Softmax	-	27,306	1,716

Table 4.7: Memory consumption and floating point operations over each network layer

Layer	**Memory Usage (MB)**	**FLOPs ($\times 10^6$)**
Input	128×100×100×3 = 3.84	-
Conv1	128×100×100×8 = 10.24	(2×3×3^2-1) ×100×100×8 = 4.24
ReLU	-	-
MaxPool1	128×50×50×8 = 2.56	-
Conv2	128×48×48×16 = 4.719	(2×8×3^2-1) ×48×48×16 = 5.272
ReLU	-	-
MaxPool2	128×24×24×16 = 1.18	-
Conv3	128×22×22×32 = 1.982	(2×16×3^2-1) ×22×22×32 = 4.445
ReLU	-	-
MaxPool3	128×11×11×32 = 0.4956	-
FC	128×1×1×2 = 0.000256	(2×3872-1) ×2 = 0.0155
Softmax	-	-

In this section, the performance of the proposed CNN model is evaluated in order to verify that MCC, sensitivity, and specificity constraints are met. The summaries of proposed model with trainable parameters, training time, prediction time, memory usage, FLOPs, working frequency, power consumption, and performance metrics are listed in Table 4.8. Additionally, the performance of proposed model is analysed based on the ROC curve as in Figure 4.5.

Table 4.8: Summaries of proposed model size, training time, prediction time, memory usage, FLOPs, frequency, power consumption, and performance metrics by the colonic polyp detection algorithm

Polyp Detection Algorithm	4-layers network		
Trainable Parameters	13,778		
Training time (s)	148	**Prediction time (s)**	1.652
Memory usage (MB)	25.017	**FLOPs ($\times 10^6$)**	13.972
Frequency (MHz)	8	**Power (mW)**	2.5488
TP	198	**TN**	192
FP	18	**FN**	12
MCC	0.8575		
ACC (%)	92.86		
SEN (%)	94.29		
PRE (%)	91.67		
SPEC (%)	91.43		
TPR	0.9429		
FPR	0.0857		
AUC	0.9870		

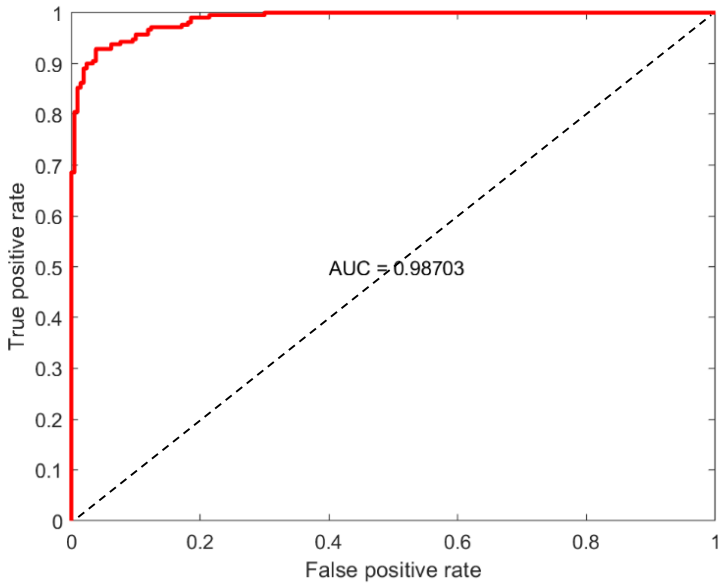


Figure 4.5: Receiver operating characteristic (ROC) graph for the proposed 4-layers model

4.4 Summary

In this research, there are two different algorithms proposed, which are categorized as phase I and phase II work. In the phase I, the proposed approach achieved state-of-the-art results by utilising the modified ResNet-50, PCA, and AdaBoost as a new combination technique to classify the polyps. Based on the preceding results and discussion, the outcome demosstrated that the CAD system manages to attain a competitive result and outperform the state-of-the-art. Meanwhile, in the phase II, finding shows that the feasibility of the CNN (4-layers network) to be implemented on the SoC with will integrated in the medical capsule robot. With the smart medical capsule robot, it can reduces the diagnosis time required by automatically detect the colonic polyp, when travelling throughout the GI tract. Moreover, in hardware implementation, the values for MCC, accuracy, sensitivity, specificity, and AUC are lowered as compared with the software implementation. This is because there is a different approach on the software and hardware implementation. On the software implementation (phase I), the modified ResNet-50 was utilised as feature extractor, and PCA is applied on the feature set before feed into the AdaBoost classifier. While

for hardware implementation (phase II), the 4-layer CNN was designed to perform classification task. This is because the memory of microprocessor is taken into consideration, the algorithm in the software implementation is not able to develop on the hardware platform.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

This chapter concludes the research findings and the contributions of the research work. Furthermore, the limitation of the proposed approach was discussed, followed by suggestions for future work.

5.1 Conclusion

Detecting and segmenting polyps become one of the challenges faced by physicians in recognizing abnormalities from endoscopic images. A robust algorithm that help to distinguish features between normal and abnormal cells might be a solution to this matter. Therefore, this study proposed a new combination of modified deep residual convolutional neural network with the PCA and ensemble learning approach for a colonic polyp classification system (phase I). Compared to the original ResNet-50, the modified ResNet-50-meta architecture achieved state-of-the-art results. Concurrently, it had a smaller size with lower complexity of the architecture. As a result, the performance of the model was improved while reducing its computational time. Additionally, an adaptive boosting-based ensemble classifier was trained on the principal component of feature extraction with the class labels (non-polyp or polyp) from the training dataset. To evaluate our model, we computed the MCC, accuracy, sensitivity, precision, and specificity. In the trial of 1,517 images from a combination of three free, publicly accessible databases, we obtained good results with 0.9819 MCC. The accuracy, sensitivity, precision, and specificity of polyp classification were 99.10%, 98.82%, 99.37%, and 99.38%, respectively. Hence, these results show that proposed algorithm is robust enough to assist in CAD it is a stable algorithm as there is no subjective bias in polyp recognition with balanced and imbalanced datasets.

Besides, the increasing of polyp miss-detected raises the case of CRC and its mortality rate, which causes by physicians' subjectivity. A smart medical capsule robot is required to detect the polyps automatically when it travels in the GI tract. However, the conventional medical capsule robot does not has the ability to detect the polyps, it only takes photos of GI tract. To develop a smart medical capsule robot for colonic polyp detection, the 4-layers network was proposed in the hardware implementation (phase II). All the essential functions of CNN (i.e., padding, convolution, ReLU, max-pooling, fully-connected, and softmax) were successfully developed into the microprocessor to show the possibility of integrating SoC into the medical capsule robot. The proposed 4-layers model achieved an accuracy of 96.35% for training and 92.86% for testing. The MCC, sensitivity, specificity, and AUC were 0.8575, 94.29%, 91.43%, and 0.9870, respectively. In sum, the suggested methods (i.e., phase I and phase II) manages to fill the research gaps by assisting the physicians to carry out the diagnosis efficiently.

5.2 Research Contributions

The research contributions are restated as follows:

a) Utilisation of AI on the CAD system to carry out the diagnosis accurately by acting as a second reader in detecting and segmenting the colonic polyps.

b) Combination of modified ResNet-50 architecture with the PCA and AdaBoost ensemble learning to detect the colonic polyps for the purpose of improving the performance of CAD tools.

c) Feasibility of implanting CNN-based model into the SoC (phase II), where the SoC is integrated in the smart medical capsule robot for the purpose of detecting the polyps automatically when the capsule robot travelling in the GI tract.

5.3 Recommendations

Based on the research findings, several further studies and works can be made to improve the performance of the proposed system. In future, these issues can be extended:

a) Phase I

The proposed CAD system still has limitation as it can misclassify some images due to the naturally irregular structure of the colon and the colour properties of the images taken by endoscopy/colonoscopy. This matter can be improved in the future by using more advanced pre-processing techniques if it is to be utilised in colonoscopy for real-time detection. Although the proposed algorithm achieved state-of-the-art results, however, the the effectiveness of the algorithm need to be tested and compared in assisting physicians in detecting polyps. Furthermore, segmentation techniques would be the way to improve the CAD system to localize the position of the polyp during real-time detection.

b) Phase II

The proposed method is promising as an automated screening approach, which the smart medical capsule robot will travels in the GI tract and detect the polyp automatically. However, the 4-layers CNN algorithm used so far is not as good as the performance in the phase I. This is because of the difficulty with this approach remains the constraint of implementing CNN layers in the hardware due to the network's complexity and the memory required. Therefore, in the future, FPGAs can be used to implement the proposed 4-layers CNN, proposed modified ResNet-50, as well as others well-known CNN models, such as AlexNet, GoogLeNet, VGG-16, and etc.

LIST OF PUBLICATION

1. W. S. Liew, T. B. Tang, C.-H. Lin, and C.-K. Lu, "Automatic colonic polyp detection using integration of modified deep residual convolutional neural network and ensemble learning approaches," *Computer Methods and Programs in Biomedicine*, vol. 206, p. 106114, Jul. 2021, doi: 10.1016/j.cmpb.2021.106114.

REFERENCES

[1] Sareena, A. Mittal, and M. Kaur, 'Computer-aided-diagnosis in colorectal cancer: A survey of state of the art techniques', in *2016 International Conference on Inventive Computation Technologies (ICICT)*, Aug. 2016, vol. 1, pp. 1–6. doi: 10.1109/INVENTIVE.2016.7823260.

[2] 'What Is Colorectal Cancer? | How Does Colorectal Cancer Start?' https://www.cancer.org/cancer/colon-rectal-cancer/about/what-is-colorectal-cancer.html (accessed May. 18, 2020).

[3] S. Patino-Barrientos, D. Sierra-Sosa, B. Garcia-Zapirain, C. Castillo-Olea, and A. Elmaghraby, 'Kudo's Classification for Colon Polyps Assessment Using a Deep Learning Approach', *Applied Sciences*, vol. 10, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/app10020501.

[4] K. Patel *et al.*, 'A comparative study on polyp classification using convolutional neural networks', *PLOS ONE*, vol. 15, no. 7, p. e0236452, Jul. 2020, doi: 10.1371/journal.pone.0236452.

[5] S. Chandan *et al.*, 'Adenoma and polyp detection rates during insertion versus withdrawal phase of colonoscopy: a systematic review and meta-analysis of randomized controlled trials', *Gastrointest Endosc*, vol. 93, no. 1, pp. 68-76.e2, Jan. 2021, doi: 10.1016/j.gie.2020.06.015.

[6] Y. Shin, H. A. Qadir, L. Aabakken, J. Bergsland, and I. Balasingham, 'Automatic Colon Polyp Detection Using Region Based Deep CNN and Post Learning Approaches', *IEEE Access*, vol. 6, pp. 40950–40962, 2018, doi: 10.1109/ACCESS.2018.2856402.

[7] D. A. Lieberman, D. G. Weiss, J. H. Bond, D. J. Ahnen, H. Garewal, and G. Chejfec, 'Use of colonoscopy to screen asymptomatic adults for colorectal cancer.

Veterans Affairs Cooperative Study Group 380', *N Engl J Med*, vol. 343, no. 3, pp. 162–168, Jul. 2000, doi: 10.1056/NEJM200007203430301.

[8] A. Nogueira-Rodríguez *et al.*, 'Deep Neural Networks approaches for detecting and classifying colorectal polyps', *Neurocomputing*, vol. 423, pp. 721–734, Jan. 2021, doi: 10.1016/j.neucom.2020.02.123.

[9] D. S. Paik *et al.*, 'Surface normal overlap: a computer-aided detection algorithm with application to colonic polyps and lung nodules in helical CT', *IEEE Trans Med Imaging*, vol. 23, no. 6, pp. 661–675, Jun. 2004, doi: 10.1109/tmi.2004.826362.

[10] N. Tajbakhsh, C. Chi, S. R. Gurudu, and J. Liang, 'Automatic polyp detection from learned boundaries', in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, Apr. 2014, pp. 97–100. doi: 10.1109/ISBI.2014.6867818.

[11] J. Stoitsis, I. Valavanis, S. G. Mougiakakou, S. Golemati, A. Nikita, and K. S. Nikita, 'Computer aided diagnosis based on medical image processing and artificial intelligence methods', *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 569, no. 2, pp. 591–595, Dec. 2006, doi: 10.1016/j.nima.2006.08.134.

[12] K. Sumiyama, T. Futakuchi, S. Kamba, H. Matsui, and N. Tamai, 'Artificial intelligence in endoscopy: Present and future perspectives', *Digestive Endoscopy*, vol. 33, no. 2, pp. 218–230, 2021, doi: 10.1111/den.13837.

[13] D. G. Lowe, 'Distinctive Image Features from Scale-Invariant Keypoints', *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, Nov. 2004, doi: 10.1023/B:VISI.0000029664.99615.94.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, 'ImageNet Classification with Deep Convolutional Neural Networks', in *Advances in Neural Information*

Processing Systems, 2012, vol. 25. Accessed: Feb. 15, 2021. [Online]. Available: https://papers.nips.cc/paper/2012/hash/c399862d3b9d6b76c8436e924a68c45b-Abstract.html.

[15] C. Szegedy *et al.*, 'Going Deeper with Convolutions', *arXiv:1409.4842 [cs]*, Sep. 2014, Accessed: Feb. 15, 2021. [Online]. Available: http://arxiv.org/abs/1409.4842.

[16] K. Simonyan and A. Zisserman, 'Very Deep Convolutional Networks for Large-Scale Image Recognition', *arXiv:1409.1556 [cs]*, Apr. 2015, Accessed: Feb. 15, 2021. [Online]. Available: http://arxiv.org/abs/1409.1556.

[17] K. He, X. Zhang, S. Ren, and J. Sun, 'Deep Residual Learning for Image Recognition', in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, Jun. 2016, pp. 770–778. doi: 10.1109/CVPR.2016.90.

[18] A. Khan, A. Sohail, U. Zahoora, and A. S. Qureshi, 'A survey of the recent architectures of deep convolutional neural networks', *Artif Intell Rev*, vol. 53, no. 8, pp. 5455–5516, Dec. 2020, doi: 10.1007/s10462-020-09825-6.

[19] Z. Pan, W. Yu, X. Yi, A. Khan, F. Yuan, and Y. Zheng, 'Recent Progress on Generative Adversarial Networks (GANs): A Survey', *IEEE Access*, vol. 7, pp. 36322–36333, 2019, doi: 10.1109/ACCESS.2019.2905015.

[20] J. Bernal *et al.*, 'Comparative Validation of Polyp Detection Methods in Video Colonoscopy: Results From the MICCAI 2015 Endoscopic Vision Challenge', *IEEE Transactions on Medical Imaging*, vol. 36, no. 6, pp. 1231–1249, Jun. 2017, doi: 10.1109/TMI.2017.2664042.

[21] H. Sharon, I. Elamvazuthi, C.-K. Lu, S. Parasuraman, and E. Natarajan, 'Development of Rheumatoid Arthritis Classification from Electronic Image Sensor Using Ensemble Method', *Sensors*, vol. 20, no. 1, Art. no. 1, Jan. 2020, doi: 10.3390/s20010167.

[22] E. Alpaydin, *Introduction to Machine Learning*, 2nd ed. Cambridge, MA, USA: MIT Press, 2009.

[23] W. Zhang, F. Zeng, X. Wu, X. Zhang, and R. Jiang, 'A Comparative Study of Ensemble Learning Approaches in the Classification of Breast Cancer Metastasis', in *2009 International Joint Conference on Bioinformatics, Systems Biology and Intelligent Computing*, Aug. 2009, pp. 242–245. doi: 10.1109/IJCBS.2009.23.

[24] X. D. Zeng, S. Chao, and F. Wong, 'Ensemble learning on heartbeat type classification', in *Proceedings 2011 International Conference on System Science and Engineering*, Jun. 2011, pp. 320–325. doi: 10.1109/ICSSE.2011.5961921.

[25] S. Pouriyeh, S. Vahid, G. Sannino, G. De Pietro, H. Arabnia, and J. Gutierrez, 'A comprehensive investigation and comparison of Machine Learning Techniques in the domain of heart disease', in *2017 IEEE Symposium on Computers and Communications (ISCC)*, Jul. 2017, pp. 204–207. doi: 10.1109/ISCC.2017.8024530.

[26] J. Wang, J. Lin, and Z. Wang, 'Efficient Hardware Architectures for Deep Convolutional Neural Network', *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 6, pp. 1941–1953, Jun. 2018, doi: 10.1109/TCSI.2017.2767204.

[27] S. S. Mapara and V. B. Patravale, 'Medical capsule robots: A renaissance for diagnostics, drug delivery and surgical treatment', *J Control Release*, vol. 261, pp. 337–351, Sep. 2017, doi: 10.1016/j.jconrel.2017.07.005.

[28] Q. Li et al., "Colorectal polyp segmentation using a fully convolutional neural network," 2017 10th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), 2017, pp. 1-5, doi: 10.1109/CISP-BMEI.2017.8301980.

[29] J. Y. Lee *et al.*, 'Real-time detection of colon polyps during colonoscopy using deep learning: systematic validation with four independent datasets', *Sci Rep*, vol. 10, no. 1, p. 8379, May 2020, doi: 10.1038/s41598-020-65387-1.

[30] R. Zhang *et al.*, 'Automatic Detection and Classification of Colorectal Polyps by Transferring Low-Level CNN Features From Nonmedical Domain', *IEEE J Biomed Health Inform*, vol. 21, no. 1, pp. 41–47, Jan. 2017, doi: 10.1109/JBHI.2016.2635662.

[31] X. Liu, Y. Li, J. Yao, B. Chen, J. Song, and X. Yang, 'Classification of Polyps and Adenomas Using Deep Learning Model in Screening Colonoscopy', in *2019 8th International Symposium on Next Generation Electronics (ISNE)*, Oct. 2019, pp. 1–3. doi: 10.1109/ISNE.2019.8896649.

[32] A. Bour, C. Castillo-Olea, B. Garcia-Zapirain, and S. Zahia, 'Automatic colon polyp classification using Convolutional Neural Network: A Case Study at Basque Country', in *2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, Dec. 2019, pp. 1–5. doi: 10.1109/ISSPIT47144.2019.9001816.

[33] H.-C. Park, Y.-J. Kim, and S.-W. Lee, 'Adenocarcinoma Recognition in Endoscopy Images Using Optimized Convolutional Neural Networks', *Applied Sciences*, vol. 10, no. 5, Art. no. 5, Jan. 2020, doi: 10.3390/app10051650.

[34] T. Ozawa, S. Ishihara, M. Fujishiro, Y. Kumagai, S. Shichijo, and T. Tada, "Automated endoscopic detection and classification of colorectal polyps using convolutional neural networks," *Therap Adv Gastroenterol*, vol. 13, p. 1756284820910659, 2020, doi: 10.1177/1756284820910659.

[35] T. Wittenberg, P. Zobel, M. Rathke, and S. Mühldorfer, 'Computer Aided Detection of Polyps in Whitelight- Colonoscopy Images using Deep Neural Networks', *Current Directions in Biomedical Engineering*, vol. 5, no. 1, pp. 231–234, Sep. 2019, doi: 10.1515/cdbme-2019-0059.

[36] W. Wang, J. Tian, C. Zhang, Y. Luo, X. Wang, and J. Li, 'An improved deep learning approach and its applications on colonic polyp images detection', *BMC Medical Imaging*, vol. 20, no. 1, p. 83, Jul. 2020, doi: 10.1186/s12880-020-00482-3.

[37] M. Liu, J. Jiang, and Z. Wang, 'Colonic Polyp Detection in Endoscopic Videos With Single Shot Detection Based Deep Convolutional Neural Network', *IEEE Access*, vol. PP, pp. 1–1, Jun. 2019, doi: 10.1109/ACCESS.2019.2921027.

[38] D. Zhou *et al.*, 'Diagnostic evaluation of a deep learning model for optical diagnosis of colorectal cancer', *Nat Commun*, vol. 11, no. 1, p. 2961, Jun. 2020, doi: 10.1038/s41467-020-16777-6.

[39] V. Vani and KM. Prashanth, 'Polyp Detection in Endoscopy Images Using Deep Learning', *EC Gastroenterology and Digestive System* 2019, vol. 6, pp.663-672.

[40] E. S. Nadimi *et al.*, 'Application of deep learning for autonomous detection and localization of colorectal polyps in wireless colon capsule endoscopy', *Computers & Electrical Engineering*, vol. 81, p. 106531, Jan. 2020, doi: 10.1016/j.compeleceng.2019.106531.

[41] J. W. Wei *et al.*, 'Evaluation of a Deep Neural Network for Automated Classification of Colorectal Polyps on Histopathologic Slides', *JAMA Netw Open*, vol. 3, no. 4, p. e203398, Apr. 2020, doi: 10.1001/jamanetworkopen.2020.3398.

[42] J. Meng *et al.*, 'Automatic detection and segmentation of adenomatous colorectal polyps during colonoscopy using Mask R-CNN', *Open Life Sciences*, vol. 15, no. 1, pp. 588–596, Jan. 2020, doi: 10.1515/biol-2020-0055.

[43] R. Xiao, J. Shi, and C. Zhang, 'FPGA Implementation of CNN for Handwritten Digit Recognition', Jun. 2020, pp. 1128–1133. doi: 10.1109/ITNEC48623.2020.9085002.

[44] S. Heller *et al.*, 'Hardware Implementation of a Performance and Energy-optimized Convolutional Neural Network for Seizure Detection', in *2018 40th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Jul. 2018, pp. 2268–2271. doi: 10.1109/EMBC.2018.8512735.

[45] M. Hügle *et al.*, 'Early Seizure Detection with an Energy-Efficient Convolutional Neural Network on an Implantable Microcontroller', in *2018 International Joint Conference on Neural Networks (IJCNN)*, Jul. 2018, pp. 1–7. doi: 10.1109/IJCNN.2018.8489493.

[46] I. Kiral-Kornek *et al.*, 'Epileptic Seizure Prediction Using Big Data and Deep Learning: Toward a Mobile System', *EBioMedicine*, vol. 27, pp. 103–111, Jan. 2018, doi: 10.1016/j.ebiom.2017.11.032.

[47] M. Odagawa *et al.*, 'A Hardware Implementation of Colorectal Tumor Classification for Endoscopic Video on Customizable DSP Toward Real-Time Computer-Aided Diagnosis System', in *2019 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2019, pp. 1–5. doi: 10.1109/ISCAS.2019.8702379.

[48] M. Khatwani, M. Hosseini, H. Paneliya, T. Mohsenin, W. D. Hairston, and N. Waytowich, 'Energy Efficient Convolutional Neural Networks for EEG Artifact Detection', in *2018 IEEE Biomedical Circuits and Systems Conference (BioCAS)*, Oct. 2018, pp. 1–4. doi: 10.1109/BIOCAS.2018.8584791.

[49] A. Jafari, S. Gandhi, S. H. Konuru, W. David Hairston, T. Oates, and T. Mohsenin, 'An EEG artifact identification embedded system using ICA and multi-instance learning', in *2017 IEEE International Symposium on Circuits and Systems (ISCAS)*, May 2017, pp. 1–4. doi: 10.1109/ISCAS.2017.8050346.

[50] M. Naresh Gowda and A. I. Rasheed, 'Hardware Implementation of Hybrid Classifier to Detect Cancer Cells', in *2017 14th IEEE India Council International*

Conference (INDICON), Dec. 2017, pp. 1–5. doi: 10.1109/INDICON.2017.8487787.

[51] C. Chen, Z. Li, Y. Zhang, S. Zhang, J. Hou, and H. Zhang, 'Low-Power FPGA Implementation of Convolution Neural Network Accelerator for Pulse Waveform Classification', *Algorithms*, vol. 13, no. 9, Art. no. 9, Sep. 2020, doi: 10.3390/a13090213.

[52] 'AdaBoost Algorithm | Quick Start Guide To AdaBoost Algorithm in Detail', *EDUCBA*, Jun. 15, 2019. https://www.educba.com/adaboost-algorithm/ (accessed Feb. 26, 2021).

[53] K. Pogorelov *et al.*, 'KVASIR: A Multi-Class Image Dataset for Computer Aided Gastrointestinal Disease Detection', Jun. 2017. doi: 10.1145/3083187.3083212.

[54] 'The Kvasir Dataset [WWW Document]'. https://datasets.simula.no/kvasir/ (accessed Feb. 26, 2021).

[55] 'Etislarib - Polyp - Grand Challenge'. https://polyp.grand-challenge.org/EtisLarib/ (accessed Apr. 12, 2020).

[56] 'Polyp - Grand Challenge', *grand-challenge.org*. https://polyp.grand-challenge.org/CVCClinicDB/ (accessed Apr. 12, 2020).

[57] A. V. Mamonov, I. N. Figueiredo, P. N. Figueiredo, and Y.-H. R. Tsai, 'Automated polyp detection in colon capsule endoscopy', *IEEE Trans. Med. Imaging*, vol. 33, no. 7, pp. 1488–1502, Jul. 2014, doi: 10.1109/TMI.2014.2314959.

[58] L. Gueye, S. Yildirim Yayilgan, F. Alaya Cheikh, and I. Balasingham, 'Automatic Detection of Colonoscopic Anomalies Using Capsule Endoscopy', Sep. 2015. doi: 10.1109/ICIP.2015.7350962.

[59] A. Nelikanti *et al.*, 'Colorectal Cancer MRI Image Segmentation Using Image Processing Techniques', *International Journal on Computer Science and Engineering (IJCSE)*, vol. 6, no. 7, pp. 280-286, Jul. 2014.

[60] B. Jeyavathana, B. Ramasamy, and A. Pandian, 'A Survey: Analysis on Pre-processing and Segmentation Techniques for Medical Images', *International Journal of Research and Scientific Innovation (IJRSI)*, Jun. 2016.

[61] 'Torch | Training and investigating Residual Nets'. http://torch.ch/blog/2016/02/04/resnets.html (accessed May. 10, 2020).

[62] C.-K. Shie, C.-H. Chuang, C.-N. Chou, M.-H. Wu, and E. Y. Chang, 'Transfer representation learning for medical image analysis', in *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, Aug. 2015, pp. 711–714. doi: 10.1109/EMBC.2015.7318461.

[63] P. F. Felzenszwalb, R. B. Girshick, D. A. McAllester, and D. Ramanan, 'Object Detection with Discriminatively Trained Part Based Models', *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2009, doi: 10.1109/TPAMI.2009.167.

[64] D. Whitley, 'Genetic Algorithms and Neural Networks', in *Genetic Algorithms in Engineering and Computer Science*, 1995, pp. 191–201.

[65] A. Ansari and A. A. Bakar, 'A Comparative Study of Three Artificial Intelligence Techniques: Genetic Algorithm, Neural Network, and Fuzzy Logic, on Scheduling Problem', in *2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology*, Kota Kinabalu, Malaysia, Dec. 2014, pp. 31–36. doi: 10.1109/ICAIET.2014.15.

[66] 'Pretrained Deep Neural Networks - MATLAB & Simulink - MathWorks United Kingdom'. https://uk.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html (accessed Apr. 5, 2020).

[67] A. Canziani, A. Paszke, and E. Culurciello, 'An Analysis of Deep Neural Network Models for Practical Applications', *arXiv:1605.07678 [cs]*, Apr. 2017, Accessed: Feb. 15, 2021. [Online]. Available: http://arxiv.org/abs/1605.07678.

[68] G. Kumar and P. K. Bhatia, 'A Detailed Review of Feature Extraction in Image Processing Systems', in *2014 Fourth International Conference on Advanced Computing Communication Technologies*, Feb. 2014, pp. 5–12. doi: 10.1109/ACCT.2014.74.

[69] R. Zhu, R. Zhang, and D.-X. Xue, 'Lesion detection of endoscopy images based on convolutional neural network features', Oct. 2015, pp. 372–376. doi: 10.1109/CISP.2015.7407907.

[70] Y. H. Liu, 'Feature Extraction and Image Recognition with Convolutional Neural Networks', *J. Phys.: Conf. Ser.*, vol. 1087, p. 062032, Sep. 2018, doi: 10.1088/1742-6596/1087/6/062032.

[71] 'Performance Comparison of Dimension Reduction Implementations — umap 0.5 documentation'. https://umap-learn.readthedocs.io/en/latest/benchmarking.html (accessed Apr. 15, 2021).

[72] S. Sivarajah, 'Dimensionality Reduction for Data Visualization: PCA vs TSNE vs UMAP vs LDA', *Medium*, Dec. 31, 2020. https://towardsdatascience.com/dimensionality-reduction-for-data-visualization-pca-vs-tsne-vs-umap-be4aa7b1cb29 (accessed Mar. 11, 2021).

[73] L. F. Sánchez-Peralta, L. Bote-Curiel, A. Picón, F. M. Sánchez-Margallo, and J. B. Pagador, 'Deep learning to find colorectal polyps in colonoscopy: A systematic literature review', *Artificial Intelligence in Medicine*, vol. 108, p. 101923, Aug. 2020, doi: 10.1016/j.artmed.2020.101923.

[74] M. K. Benkaddour and A. Bounoua, 'Feature extraction and classification using deep convolutional neural networks, PCA and SVC for face recognition',

Traitement du signal, vol. 34, no. 1–2, pp. 77–91, Oct. 2017, doi: 10.3166/ts.34.77-91.

[75] H. Lu, L. Yang, K. Yan, Y. Xue, and Z. Gao, 'A cost-sensitive rotation forest algorithm for gene expression data classification', *Neurocomputing*, vol. 228, pp. 270–276, Mar. 2017, doi: 10.1016/j.neucom.2016.09.077.

[76] M. F. I. Ibrahim and A. A. Al-Jumaily, 'PCA indexing based feature learning and feature selection', in *2016 8th Cairo International Biomedical Engineering Conference (CIBEC)*, Dec. 2016, pp. 68–71. doi: 10.1109/CIBEC.2016.7836122.

[77] M. Zhu *et al.*, 'PCA and Kernel-based extreme learning machine for side-scan sonar image classification', in *2017 IEEE Underwater Technology (UT)*, Feb. 2017, pp. 1–4. doi: 10.1109/UT.2017.7890275.

[78] M. Gupta and B. Gupta, "A Comparative Study of Breast Cancer Diagnosis Using Supervised Machine Learning Techniques," 2018 Second International Conference on Computing Methodologies and Communication (ICCMC), 2018, pp. 997-1002, doi: 10.1109/ICCMC.2018.8487537.

[79] M. Amrane, S. Oukid, I. Gagaoua and T. Ensarİ, "Breast cancer classification using machine learning," 2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting (EBBT), 2018, pp. 1-4, doi: 10.1109/EBBT.2018.8391453.

[80] K. K. Jena, S. Kumar Bhoi, D. Mohapatra, C. Mallick and P. Swain, "Rice Disease Classification Using Supervised Machine Learning Approach," 2021 Fifth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2021, pp. 328-333, doi: 10.1109/I-SMAC52330.2021.9641054.

[81] 'classification - Adaptive Boosting vs. SVM', *Cross Validated*. https://stats.stackexchange.com/questions/111654/adaptive-boosting-vs-svm (accessed May. 22, 2020).

[82] J. P. Vink and G. de Haan, ‘Comparison of machine learning techniques for target detection’, *Artif Intell Rev*, vol. 43, no. 1, pp. 125–139, Jan. 2015, doi: 10.1007/s10462-012-9366-7.

[83] H. Lu, Y. Meng, K. Yan, and Z. Gao, ‘Kernel Principal Component Analysis Combining Rotation Forest Method for Linearly Inseparable Data’, *Cognitive Systems Research*, vol. 53, Feb. 2018, doi: 10.1016/j.cogsys.2018.01.006.

[84] S. Pang, Y. Zhang, M. Ding, X. Wang, and X. Xie, ‘A Deep Model for Lung Cancer Type Identification by Densely Connected Convolutional Networks and Adaptive Boosting’, *IEEE Access*, vol. 8, pp. 4799–4805, 2020, doi: 10.1109/ACCESS.2019.2962862.

[85] L. Shi *et al.*, ‘The MAQC-II project: a comprehensive study of common practices for the development and validation of microarray-based predictive models’, *Nat Biotechnol*, vol. 28, no. 8, pp. 827–838, Aug. 2010.

[86] The SEQC/MAQC-III Consortium, ‘A comprehensive assessment of RNA-seq accuracy, reproducibility and information content by the Sequencing Quality Control Consortium’, *Nat Biotechnol*, vol. 32, no. 9, pp. 903–914, Sep. 2014, doi: 10.1038/nbt.2957.

[87] L. Shi *et al.*, ‘The MicroArray Quality Control (MAQC)-II study of common practices for the development and validation of microarray-based predictive models’, *Nat Biotechnol*, vol. 28, no. 8, pp. 827–838, Aug. 2010, doi: 10.1038/nbt.1665.

[88] W. Zhang *et al.*, ‘Comparison of RNA-seq and microarray-based models for clinical endpoint prediction’, *Genome Biology*, vol. 16, no. 1, p. 133, Jun. 2015, doi: 10.1186/s13059-015-0694-1.

[89] D. Chicco and G. Jurman, ‘The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation’, *BMC Genomics*, vol. 21, no. 1, p. 6, Jan. 2020, doi: 10.1186/s12864-019-6413-7.

[90] G. Lacey, G. W. Taylor, and S. Areibi, 'Deep Learning on FPGAs: Past, Present, and Future', *arXiv:1602.04283 [cs, stat]*, Feb. 2016, Accessed: Aug. 15, 2021. [Online]. Available: http://arxiv.org/abs/1602.04283.

[91] T. H. Vu, R. Murakami, Y. Okuyama, and A. Ben Abdallah, 'Efficient Optimization and Hardware Acceleration of CNNs towards the Design of a Scalable Neuro inspired Architecture in Hardware', in *2018 IEEE International Conference on Big Data and Smart Computing (BigComp)*, Jan. 2018, pp. 326–332. doi: 10.1109/BigComp.2018.00055.

[92] Z. Liao, R. Gao, C. Xu, and Z.-S. Li, 'Indications and detection, completion, and retention rates of small-bowel capsule endoscopy: a systematic review', *Gastrointestinal Endoscopy*, vol. 71, no. 2, pp. 280–286, Feb. 2010, doi: 10.1016/j.gie.2009.09.031.

[93] S. J. Heitman, P. E. Ronksley, R. J. Hilsden, B. J. Manns, A. Rostom, and B. R. Hemmelgarn, 'Prevalence of adenomas and colorectal cancer in average risk individuals: a systematic review and meta-analysis', *Clin Gastroenterol Hepatol*, vol. 7, no. 12, pp. 1272–1278, Dec. 2009, doi: 10.1016/j.cgh.2009.05.032.

[94] D. H. Kim, P. J. Pickhardt, and A. J. Taylor, 'Characteristics of advanced adenomas detected at CT colonographic screening: implications for appropriate polyp size thresholds for polypectomy versus surveillance', *AJR Am J Roentgenol*, vol. 188, no. 4, pp. 940–944, Apr. 2007, doi: 10.2214/AJR.06.0764.

[95] Y. Ren, Y. Chen, L. Lu, and J. Zhao, 'Novel 3D Radiomic Features for Computer-Aided Polyp Detection in CT Colonography', *IEEE Access*, vol. 6, pp. 74506–74520, 2018, doi: 10.1109/ACCESS.2018.2874803.

[96] H. Yuan, Q. Yu, Y. Zhang, Q. Yu, Q. Zhang, and W. Wang, 'Ultrasound Radiomics Effective for Preoperative Identification of True and Pseudo Gallbladder Polyps Based on Spatial and Morphological Features', *Frontiers in Oncology*, vol. 10, 2020, doi: 10.3389/fonc.2020.01719.

[97] H. Shakir, Y. Deng, H. Rasheed, and T. M. R. Khan, 'Radiomics based likelihood functions for cancer diagnosis', *Sci Rep*, vol. 9, no. 1, p. 9501, Jul. 2019, doi: 10.1038/s41598-019-45053-x.

[98] S. A. Azer, 'Challenges Facing the Detection of Colonic Polyps: What Can Deep Learning Do?', *Medicina (Kaunas)*, vol. 55, no. 8, p. 473, Aug. 2019, doi: 10.3390/medicina55080473.

[99] M. Nagendran *et al.*, 'Artificial intelligence versus clinicians: systematic review of design, reporting standards, and claims of deep learning studies', *BMJ*, vol. 368, p. m689, Mar. 2020, doi: 10.1136/bmj.m689.

[100] G. Urban *et al.*, 'Deep Learning Localizes and Identifies Polyps in Real Time With 96% Accuracy in Screening Colonoscopy', *Gastroenterology*, vol. 155, no. 4, pp. 1069-1078.e8, Oct. 2018, doi: 10.1053/j.gastro.2018.06.037.